symmetry

MDPI

Article

Adaptive Mechanism Model for the Prevention of SLA Violation in the Context of COPD Patient Monitoring

Konan-Marcelin Kouamé *, Hamid Mcheick and Hicham Ajami

Department of Computer Sciences and Mathematics, University of Québec at Chicoutimi, Chicoutimi, QC G7H 2B1, Canada; hamid_mcheick@uqac.ca (H.M.); hicham.ajami1@uqac.ca (H.A.)
* Correspondence: konan-marcelin.kouame1@uqac.ca

Received: 20 August 2020; Accepted: 17 September 2020; Published: 22 September 2020

Abstract: In this paper, we introduce a new kind of Service Level Agreement(SLA) Template to better control dynamically quality of medical monitoring platform service. Our approach is based on Health care system and Health Information Technology (HIT) research area, specifically the field of telemonitoring system for patients who suffer from chronic obstructive pulmonary disease (COPD). According to WHO statistics, COPD is the third leading cause of death worldwide. To this end, several solutions or platforms exist today to monitor COPD. Most of these platforms manage large volume of patient data. This can bring about quality and lost data problems. To address these issues, control mechanisms must be proposed and designed to improve the quality of service (QoS) on these platforms. A platform with continuously monitored QoS can save patients' lives and reduce data quality risk. In this article, we propose an ontology that uses SLAs data from COPD monitoring platforms with dynamic data from a patient context. We dynamically calculate the number of patient data incidents and the number of service request incidents from two dynamic contexts: SLA and the patient context. If the number of incidents is higher than what is expected in the SLA, then alerts are sent to the interface parties in real time. Finally, the contribution of this article is the proposed virtual SLA template to better control SLA violation and improve quality of medical monitoring platforms services.

Keywords: health care system; adaptation; SLA; violation; adaptive SLA template; ontology; context-aware; virtual; dynamic adaptation; context-aware application

1. Introduction

During the last few decades, ubiquitous computing, particularly Health Care System and Health Information Technology (HIT), commonly referred to as pervasive/ubiquitous systems, has impacted the world of research. This revolution has materialized through many context-aware applications that have become important in different fields of research in general, and particularly in the medical field. Health Information Technology (HIT) is health technology, particularly information technology, applied to health and health care. It supports health information management across computerized systems and the secure exchange of health information between consumers, providers, payers, and quality monitors. Service level agreement (SLA) was introduced in the 1980. SLA has been used in various fields such as telecommunications, call centers, security, cloud computing and health systems. In effect, remote medical monitoring platforms manipulate large volumes [1] of patient data, and the risk of data loss or poor data quality is real. This situation can be dangerous for chronic obstructive pulmonary disease (COPD) patients in extremely exacerbated cases. Moreover, quality of service (QoS) is important in the field of monitoring systems because context-aware applications must constantly adapt their services to the context [2]. The context changes dynamically, and meeting QoS requirements is a challenge. Tamura et al. [2] explained that the necessity to meet the quality level requirements of the platforms in variable execution contexts is a major problem. Context-aware applications must constantly adapt their services to the changing context. The question is how to dynamically monitor the service level agreement (SLA) during an adaptation activity in which the context can be changed. Abawajy et al. [3] state that an SLA document includes negotiable information and unnegotiable constraints. In SLA models, we can identify standard types of metric groupings, such as platform performance metrics, hardware performance metrics, and software performance metrics. Khattak argues that context quality of service (CQoS) errors [4], which can occur at any stage of the adaptation process, can have an impact on the quality of service provided by contextual applications. Khattak [4] et al., declared that the contextual data fusion is performed from several sensor sources. However, these sensors may have problems of erroneous reading and connection failure. As a result of data acquisition errors and failures, sensors can provide erroneous data. At the inference level, the algorithms on which the calculations are based may be erroneous. All these situations can cause violations of the SLAs between the providers of context-aware applications and users or customers. It is therefore important to have mechanisms to continuously control and improve the quality of service (QoS) on these platforms. Research associated with SLAs for cloud computing is increasing, but in the context of health monitoring, especially in the context of health surveillance, the dynamic QoS control mechanisms of monitoring platforms are not fully addressed [4]. In this article, we propose an ontology that uses SLA information from COPD monitoring platforms with dynamic data from the patient context. The purpose of this article is to propose a dynamic mechanism for evaluating SLA violations. The solution allows us to retrieve the main items from SLA based XML documents and then retrieve the COPD patient context data dynamically from a COPD SLA ontology. These data retrieved in real time allow the calculation of key performance indicator (KPI) targets based on service level objective (SLO). The adaptive SLA model displays the result of SLA control. This model is available on supplier and consumer interfaces in real time. The information on the interface for SLA control changes dynamically depending on the context-aware system data and the SLA document data. The SLA parties can dynamically control their KPI targets. Our solution can be also deployed on different COPD patient monitoring systems in the application layer to integrate quality control mechanisms. This paper presents our adaptive mechanism model to better control SLA violations in the context of monitoring COPD patients. The structure of the article is organized as follows: Section 2 describes related works, Section 3 presents the proposed model, Section 4 describes the design of the virtual adaptive template SLA and its implementation, and Section 5 presents the conclusion and future works.

2. Related Works

Shah et al. [5] discuss the concept of QoS for the health care system. The authors explain that these systems manipulate large volumes of data and declare that quality control mechanisms are needed to improve the QoS. Shah et al. also [5] explain the importance of adapting the SLA approaches of the systems to patient monitoring. The quality of transmission for data collected from patients is a very promising area of research in the health field according to Khan et al. [4]. Besides, body area networks (BANs) are an emerging field with applications for monitoring vital signs in addition to other contextual information. Khan et al. [4] proposed a solution for the transmission of patient data. The QoS routing protocol for reliability sensitive data in hospital body area networks [4] is an approach that ensures the quality of the transmission of data collected from patients, but it does not guarantee transmission quality when there are a large number of patients to be monitored. Furthermore, this approach does not provide an agreement between the patient and the clinic about monitoring the QoS. The requirements of patients are not addressed in the definition of this approach. In the same concept of service quality approach, Kouame et al. [6] and Abawajy et al. [3] indicate that an SLA model is a document that defines negotiable service level objectives (SLOs) and constraints. Other research projects have used SLA to improve system QoS. This is the case of the SLA predictive monitoring project [7]. According to Comuzzi et al. [7], this research project presented a new definition of the reliability of SLA forecasts, which included terms related to the variability of the service provision scenario in which a prediction is made. The predictions were based on SLOs, but this later project was limited by the absence of dynamic control of the SLA. Another research project [7] proposed an SLA-based healthcare big data analysis and computing in cloud network by combining big data analysis techniques with data collected from patients. In addition, according to Sahoo [8], an algorithm is proposed to prioritize the patient emergency work. Despite the contribution of this research project to prioritizing patient emergency work, this project has limitations, such as non-dynamic SLA and absence of an SLA violation control mechanism. The research project name [9], *Cloud SLA terms analysis based on ontology* focused on the problem of SLA and QoS violation in the context of services offered by the cloud. In the context of service offered by providers, this project allows subscribing consumers to monitor SLA violation, however, the data for calculating the metrics were not extracted from a patient context and no adaptation mechanism was implemented. The authors even indicated the limitations of this approach [7] and declared: *"For future work, we opt to enhance the process by automatically predicting SLA violation using context-aware systems"*. Finally, in the same vein, the *AISel* project [8] aimed to help emergency physicians or practitioners to have relevant information and knowledge about COPD patients to better monitor their health status. This project proposed a virtual organization (VOS) to share knowledge between patient and practitioner actors in compliance with data protection laws. The organization based on SLA, a multi-agent system, explained in [10] that: *"Conceptual definitions of SLA parameters, metrics, and economic values, as well as resource characteristics, are given in an OWL-DL ontology W3C"*. The *MORF* project of Luke et al. [11], was a mobile patient monitoring system with two vital parameters: SAO2(Saturation Oxygen) and HR(Heart Rate). This monitoring system seamlessly integrated vital data from different health services' sensors to identify a patient's condition. The system could then take appropriate measures for timely support. It was an ontology-based system that generated alerts based on the values of vital parameters. This system used ontology-based reasoning, but the QoS aspect of the system was not addressed. Finally, Labidi et al. [12] worked on an ontology-based SLA project. The objective of this project was to help users to better understand the terms of the SLA as part of the monitoring of its violation. Labidi et al. [12] explained: *"We describe the cloud SLA modelling and analyzing approach that assists consumers to automatically analyze the terms of various SLA documents"*. Their approach was based on static data. They [12] plan to integrate the SLA management process by automatically predicting SLA violation using context-aware systems.

Table 1 describes relevant existing projects of SLA violation.

Table 1. Abstract of existing projects summarizing existing works.

References	Project Description	Limits
[1]	The transmission quality for data collected from patients	Not guarantee of data quality when there are a large number of patients to be monitored. No agreement to monitor QoS
[5]	Virtual organization (VOS) to share knowledge	No agreement to monitor QoS. No SLA violation control mechanism using context-aware system
[3,6,7,9]	Importance of SLA for QoS improvement. Prevention of SLA violation	No dynamic SLA, and no SLA violation control mechanism.
[10–12]	Ontology-based reasoning	QoS aspects of the system are not addressed; absence of the dynamic aspect

3. Proposed Model

3.1. SLA Concepts & Approach

Telemonitoring is an essential activity, not only to improve the quality of life of COPD patients, but also to help emergency physicians make monitoring decisions with real-time data. Today, we are experiencing the emergence of several remote monitoring systems for COPD patients. As these systems are extremely important to the health of COPD patients and the quality of care, QoS must be assessed [4,5]. Our proposal is based on a virtual dynamic SLA template that monitors the patient context and updates the violation control interface. According to [13], the SLA is the main document that guarantees the QoS provided by a computer system. It defines a service level agreement between the IT service provider and the consumer of these IT services. Kritikos et al. [14] argued that an SLA comprises a set of service levels (SLs) explicating the different performance behaviors that a service can exhibit. The different activities in the life cycle of an SLA are description, matchmaking, negotiating, monitoring, assessment, settlement, and archiving [15]. Different languages are used to design and implement the SLA. The structure of the SLA document in Web Service Agreement(WSA) format allows us to translate the SLA document into an XML file. In this XML file, we can easily extract the data with another XML-compatible language. Our approach, described in Figure 1, is based on a WSA format in an XML-compatible language. Figure 1 describes our approach to design dynamic control mechanisms for SLA in a COPD monitoring context. Indeed, our approach consists of the four steps described below.

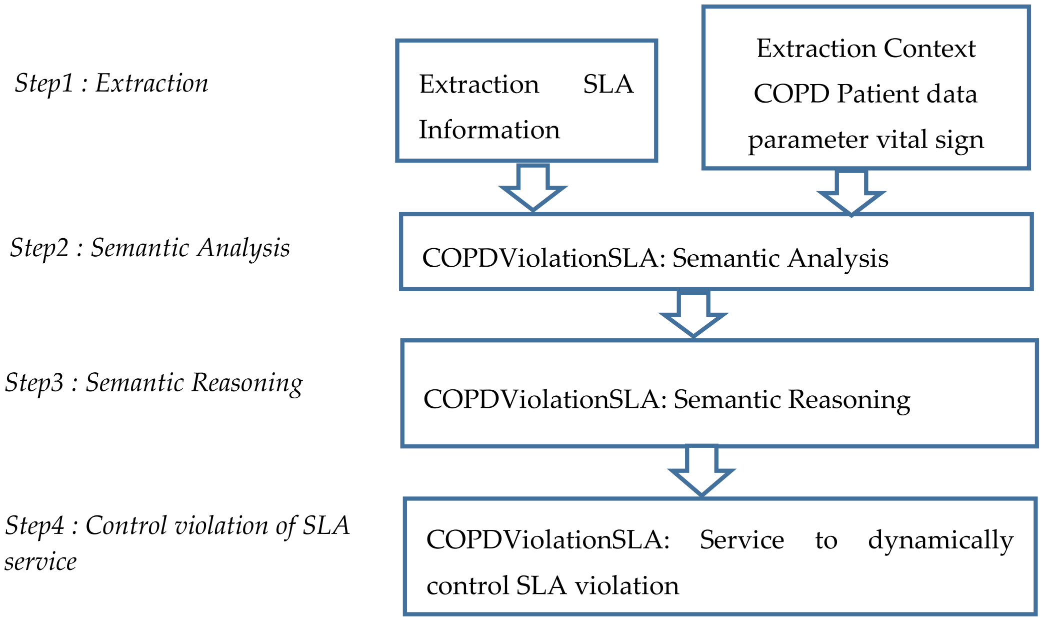

Step1 : Extraction
Extraction SLA Information
Extraction Context COPD Patient data parameter vital sign
Step2 : Semantic Analysis
COPDViolationSLA: Semantic Analysis
Step3 : Semantic Reasoning
COPDViolationSLA: Semantic Reasoning
Step4 : Control violation of SLA service
COPDViolationSLA: Service to dynamically control SLA violation

Figure 1. Description of the different steps of research approach and design of the virtual service level agreement (SLA) approach.

Steps of the Approach

(i) Step 1: Extraction Data and Rules

Data extraction consists of extracting information from the COPD patient context (vital patient parameters sign) and the SLA (metric or KPI-targets). On the SLA side, the information is the SLA items in WSA format and the rules. On the COPD side, the information source is the real-time database.

(ii) Step 2: Semantic Analysis

This step consists of analyzing the items, proposing rules based on information from the SLA side and on the context-aware system side. The information on the vital parameters data allows for populating in real time the COPDViolationSLA ontology model.

(iii) Step 3: Semantic Reasoning

The COPD rules and patient data are used in the reasoning process to generate the dynamic control services and populate the adaptive and virtual template.

(iv) Step 4: Control Violation of SLA Service

Depending on the period chosen by the user, it is possible to control the SLA violation in real time. It is also possible for the user to set the control period, for example, every hour. Whenever the patient context changes, the control mechanisms based on COPD violation SLA ontology change the information in the template.

3.2. Dynamic Control Architecture for COPD Diseases Based on SLA

The architecture of COPD adaptive control is described in four layers.

Figure 2 presents the architecture of the proposed model.

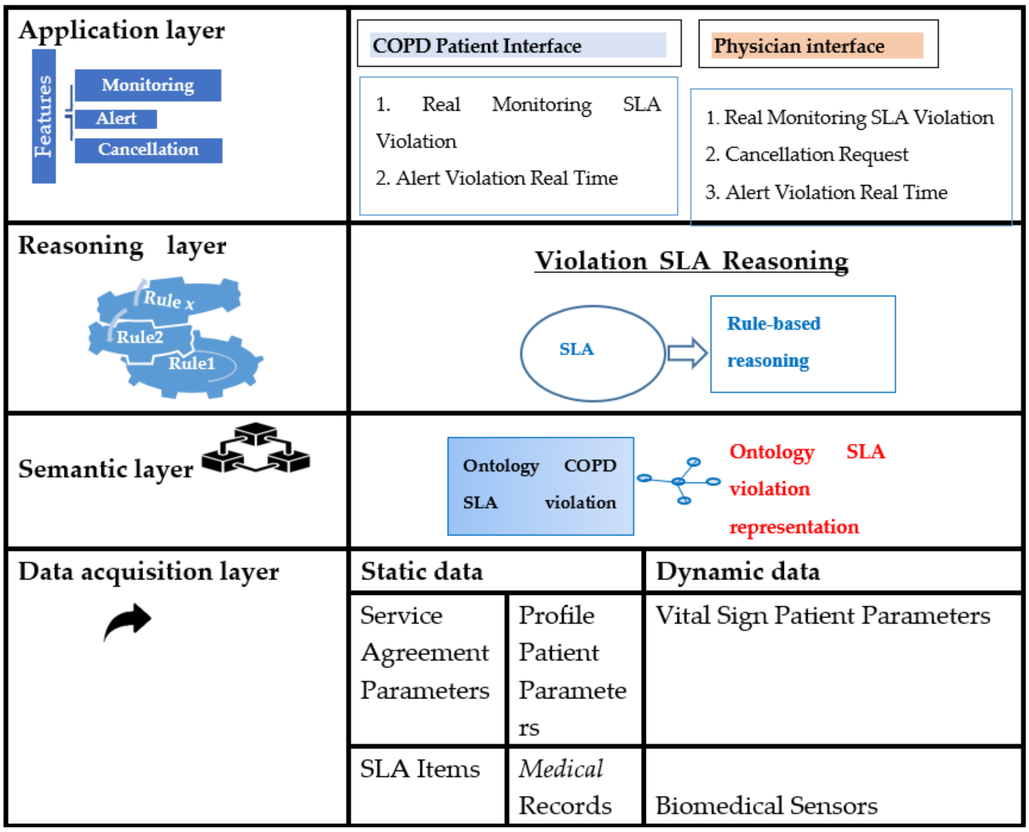


Figure 2. Architecture of proposed model.

The application layer presents the emergency doctor, the patient, emergency reception, and triage interface.

3.2.1. Application Layer

The application layer provides two categories of services (i) administrator service and (ii) functional services for the control of the SLA. The administrator service allows the emergency physician to create or delete vital sign parameters (FIEV, SAo2, HR, temperature) with their respective limits, triage agent profiles, and COPD patient profiles in the system. The functional services for SLA monitoring are available in the different interfaces described below. The control interfaces allow for the dynamic monitoring of the SLA according to the period chosen by the user.

(i) Emergency Interface

The emergency physician's interface allows the SLA control by period. This interface can also receive alerts depending to the patient and SLA context. Only the emergency physician can cancel the SLA.

(ii) Triage Agent Interface

The agent interface helps to perform triage activity by using triage dashboard at the emergency reception desk. This dashboard is dynamically updated.

(iii) Patient Interface

This interface can receive alerts depending to the patient and SLA context.

3.2.2. Reasoning Layer

The reasoning engine is mainly based on rules, the KPI_TARGET value extracted from the SLA, and the income data sensor. The reasoning process consists of four phases: the phase of identification of the metric items or KPI target, the detection incident phase result from the comparison of the vital data of the four parameters collected on COPD patients, the implementation phase of the SLA violation control algorithm, and the phase of populate the SLA virtual template. These rules are important for the SLA violation evaluation mechanism. The two metrics of the dynamic SLA violation monitoring mechanism in the context of COPD patient are: the number of incidents from patient data, numberIncidentData, and the number of incidents of unsuccessful patient request, NumBeRequestSend. In effect, if the number of incidents is greater than the metric defined in the KPI-Target, there is a violation of the SLA. The rules for the SLA evaluation mechanisms are important for dynamically triggering the control of the SLA according to the context-aware data and SLA document information in WSA format.

(i) Detection Incident Phase

The information extracted from the SLA can be grouped into two parts: Content items and KPI-TARGETs. The KPI-TARGETs participate in the dynamic control mechanisms of the SLA. The other content items populate the virtual adaptive SLA template. KPI_IncidentDataValue and KPI_IncidentRequestValue are important items for the mechanism of SLA control violations.

To design these rules, we consider that for every vital sign parameter, there is an interval with a minimum and a maximum value, in which the value of the sensor is a data incident.

Table 2 describes these incident rules.

Table 2. Incident Rules.

Incident Data	Rules based on Ontology Web Language Description Logics (OWL/DL/SWRL)
IncidentTemperature	TemperatureSensorData(?tsd) ^ hasCurrentValue(?tsd, ?cv) ^ hasRangeMin(?tsd, ?Min) ^ hasRangeMax(?tsd, ?Max) ^ swrlb:greaterThanOrEqual(?cv, ?Min) ^ swrlb:lessThanOrEqual(?cv, ?Max) –> IncrementAlarmLevel(?tsd, AlarmLevelIncrement)
IncidentFiev	TemperatureSensorData(?tsd) ^ hasCurrentValue(?tsd, ?cv) ^ hasRangeMin(?tsd, ?Min) ^ hasRangeMax(?tsd, ?Max) ^ swrlb:greaterThanOrEqual(?cv, ?Min) ^ swrlb:lessThanOrEqual(?cv, ?Max) –> IncrementAlarmLevel(?tsd, AlarmLevelIncrement)
IncidentHrRule	HeartRateSensorData(?hsd) ^ hasCurrentValue(?hsd, ?cv) ^ hasRangeMin(?hsd, ?Min) ^ hasRangeMax(?hsd, ?Max) ^ swrlb:greaterThanOrEqual(?cv, ?Min) ^ swrlb:lessThanOrEqual(?cv, ?Max) –> IncrementAlarmLevel(?spo2, AlarmLevelIncrement)
IncidentSAo2Rule	SAO2SensorData(?sao2) ^ hasCurrentValue(?sao2, ?cv) ^ hasRangeMin(?sao2, ?Min) ^ hasRangeMax(?sao2, ?Max) ^ swrlb:greaterThanOrEqual(?cv, ?Min) ^ swrlb:lessThanOrEqual(?cv, ?Max) –> IncrementAlarmLevel(?sao2, AlarmLevelIncrement)

If IncidentDataNumber > KPI_IncidentDataValue, then the data quality SLA violation alarm is triggered.

Figure 3 presents the violation SLA based on incident data module.

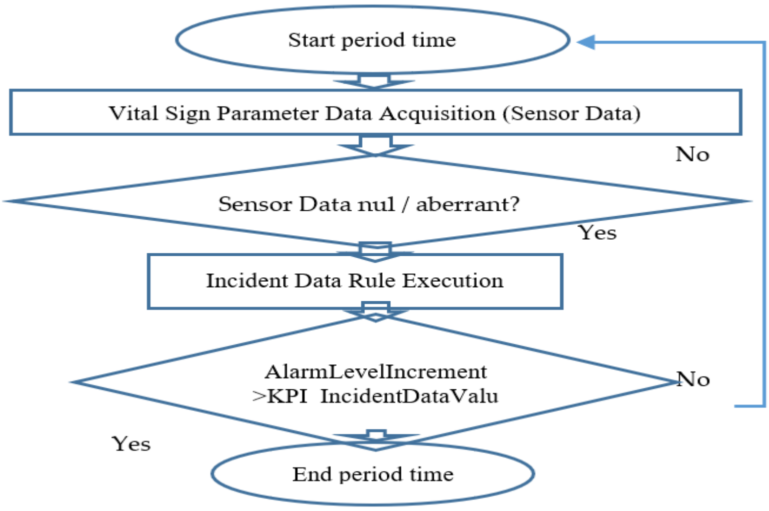

Start period time
Vital Sign Parameter Data Acquisition (Sensor Data)
No
Sensor Data nul / aberrant?
Yes
Incident Data Rule Execution
AlarmLevelIncrement
>KPI IncidentDataValu
No
Yes
End period time

Figure 3. SLA data quality violation algorithm.

There are two approaches to SLA violation control: the data incident approach and the request incident approach. For data incidents, the reasoning is performed as soon as the following data are entered into the dynamic database: FIEVSensorData, HearateSensorData, and TemperatureSensorData. For each piece of data of the parameters, the system checks through each rule to determine if it is an aberrant or erroneous value. If it is an aberrant or erroneous value, then it is classified as a data incident. The system increments the number of incident data and compares this number to the metric KPI_IncidentValue extracted from the SLA.

Concerning request incidents, the incident counter (IncidentRequestNumber) is calculated as the difference between the number of incidents (RequestSendNumber) sent to the emergency physician by patients and the number of requests (RequestReceiveNumber) confirmed or received by the emergency physician during period. The system then compares the KPI_IncidentRequestValue extracted from the SLA to the IncidentRequestNumber. If IncidentRequestNumber > KPI_IncidentRequestValue, then a QoS SLA violation alert is triggered. All these mechanisms are implemented in the SLA violation algorithm. This algorithm represents the core of the system's reasoning. Below, Figure 4 describes the algorithm of SLA violation mechanisms based on incident requests.

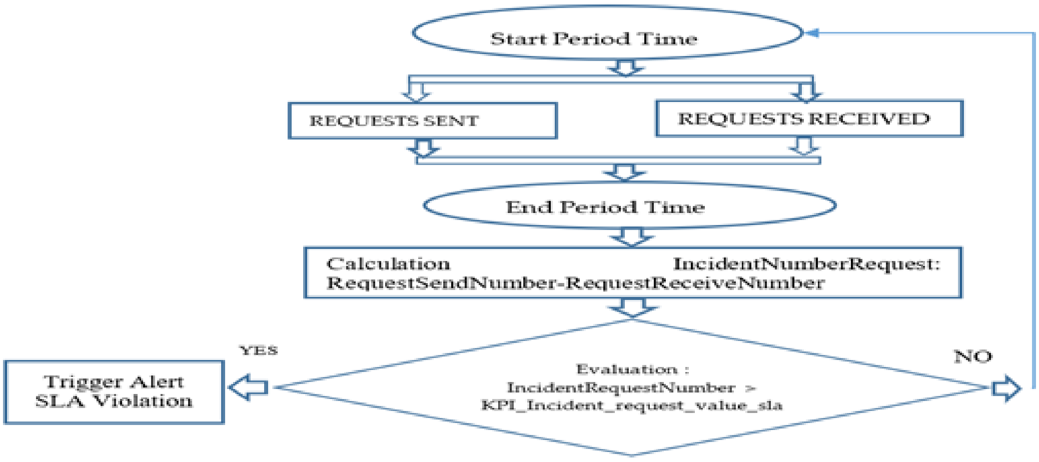


Figure 4. SLA quality service violation algorithm.

(ii) Populate Phase

The populate phase consists of displaying the results of the evaluation algorithm for SLA violation to the virtual template. These results are therefore accessible from the interface in a dynamic mode.

3.2.3. Semantic Layer: Development of COPDViolationSLAOntology Domain

Ajami and Mcheick [16] explain that the semantic approach is a formalism often used to interpret complex information. With the protégé Semantic Web Reasoning Language(SWRL) tool, it is possible to perform queries based on concepts or classes [16]. Ajami and Mcheick [16] declared that: "*Ontology is considered to be one of the richest semantic structures for facilitating the representation, integration, and reasoning of knowledge*."

(i) Structure of COPDViolationSLAOntology Module

Figure 4 shows the core of the COPDViolationSlaOntology structure. Each component of this structure is an ontology that participates in the implementation of the dynamic evaluation mechanisms of SLA. Each component can be reused to enrich other SLA ontologies. The structure is modular. Each module is described in detail in the following sections. This structure can be linked to the standard COPD ontology, i.e., COPD ontology from the patient COPD Ontology component. COPDViolationSLa Ontology is therefore not an independent ontology. It is a modular ontology that is reusable and extensible with the SLA Ontology Standard.

Figure 5 shows an overview of the core structure of COPDViolationSLaOntology.

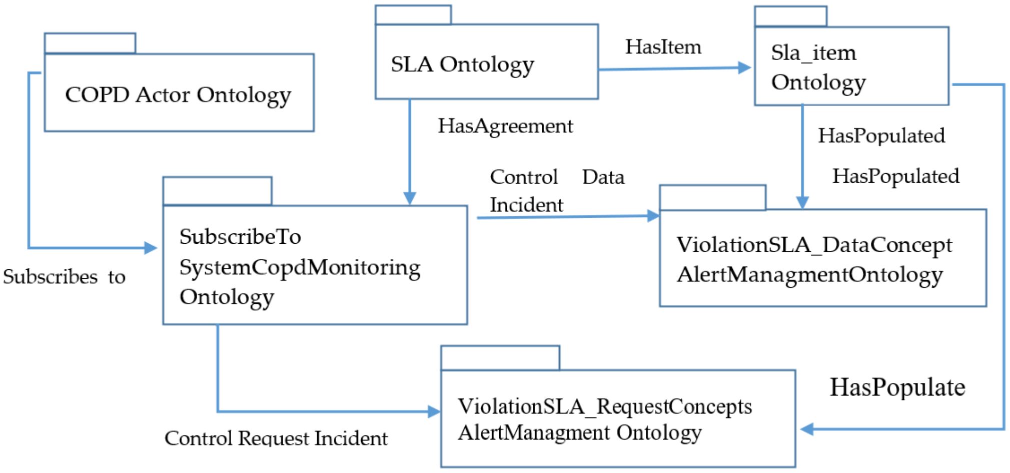


Figure 5. Core structure of COPDViolationSLaOntology.

(ii) COPDPatientOntology Module

The objective of this ontology in the design of SLA control mechanisms is the dynamic collection of COPD patient data. The data come from the vital signal parameters: Fiev, Heartrate, Sao2, and Temperature. The Object Property and Datatype relationships of this ontology are described in Figure 4. Only the main relationships that participate in the dynamic control mechanisms of the SLA are described in Figure 4.

The object property relationships that participate in the SLA violation evaluation mechanisms are hasTemperatureSensorData, hasHeartRateSenSorData, hasSAO2SensorData, and hasFIEVSensorData. Indeed, these relationships allow for the collection of temperature, heart rate, oxygen saturation, and FIEV data, respectively, from a sensor installed on each COPD patient. The collected data are evaluated according to the rules defined in the reasoning. During this period, if the collected data do not satisfy the rules policies, then the reasoning engine increments the SLA violation incident counter.

When the counter reaches the threshold defined in the KPI-Target Data incident, then the inference engine allows the SLA violation alert to be triggered.

Our solution approach is based on an adaptation mechanism. In effect, the type of alert sent to the parties for the SLA monitoring changes according to the nature of the incident detected by the patient context, for example by time period, if the patient context provides erroneous data, the alert system will generate a data incident.

In another period, if it is the service incidents that are identified, the alert will generate service incidents and not data incidents. Finally, depending on the nature of the incident, the system will propose dynamically the appropriate alert and services.

Table 3 presents some properties of the ontology patient module. Figure 6 describes COPD Patient Ontology Module

Table 3. Some properties of patient ontology module.

Object Property	Domain	Range	Datatype Property
hasPatientPersonalProfil	PatientCopd	PatientPersonalProfile	hasName
hasEmergencyDoctor	PatientCopd	EmergencyDoctor	hasContact
useSensor	PatientCopd	Sensor	hasSensoirId
hasAlarmViolationSlaManagementProfile	PatientCopd	AlarmViolationSlaManagementProfile	hasaLarmlevel
hasTemperatureSensorData	PatientCopd	TemperatureSensorData	hasTemperatureSensorData

Figure 6 described patient ontology.

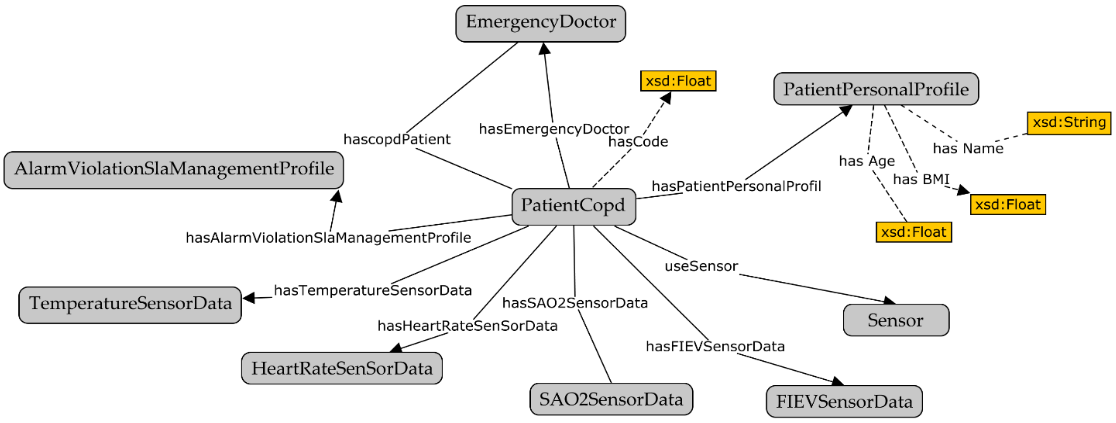


Figure 6. COPD Patient Ontology Module.

Patient ontology module is structured into two main branches: vital parameters, psychological profile parameters, and patient personal information. The vital parameter sign represents the core of the evaluation mechanism. The SLA control rules are mainly based on the values of these parameters. For example, if the current value of the temperature does not belong to minimum and maximum temperature value, then there is a data incident. The SLA control counter is then incremented. If the counter value is beyond the KPI-TARGET limits, then the SLA control mechanism triggers the SLA violation alert according to the period chosen by the user. All signatory actors are automatically informed.

(iii) SLAItemsOntology Module

This ontology aims to provide information about the SLA context. For this purpose, it only addresses the description of the different items of the SLA. The items that are mainly involved in the mechanisms of control SLA are KPI-TARGET, Services and SLO. The Object Property and Datatype Property that characterize them are represented in the main table. The hasKPI_Expression_IncidentRequest and hasKPI_Expression_IncidentData relationships identify the incident levels discussed with the various parties signing the SLA. Indeed, the value of the data incident counter is compared with the KPI-TARGET defined in the SLO expression related to the data: KPI_Expression_IncidentData. The same approach is true for the request incident counter value, which is also compared with the SLO expression related to the incident requests: KPI_Expression_IncidentRequest.

Table 4 presents some properties of SLA items ontology.

Table 4. Description of some properties of SLAItemsOntology module.

Object Property	Domain	Range	Datatype Property
HasSLOItems	SLA_Item	SLO	hasIncidetn_Data_value
isexpressedas	SLO	KPI-TARGET	hasIncident_Request
hasServiceItems	SLA_Item	Service	hasServiceId
hasConstraintItems	SLA_Item	Constraints	hasConstraintId
hascondition	SLA_Item	TemperatureSensorData	hasTemperatureSensorDataMin
hasTermsItems	SLA_Item	Terms	hasTerm_Id
Isrealatedto	Service	SLO	hasSLO_Name

The details of the SLAItemsOntology module are described in Figure 7 below.

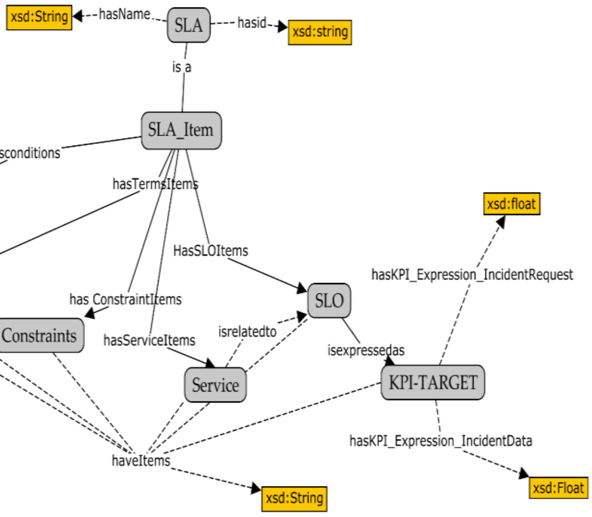


Figure 7. SLAItemsOntology module.

(iv) Violation Data Concepts and Alarm Management module

The purpose of this ontology is to record violations information and then populates the information to the template dynamically. The items or concepts that mainly participate in the SLA control mechanisms are patient and violation Sla Data Concepts. The hasAlarmViolationSlaManagementProfile and IncrementAlarmLevel links respectively associate the incident to the patient concerned and increment the incident counter to compare it to the KPI-TARGET extracted from the SLA. The hasAcquisitionRate relationship allows the emergency physician to define data collection periods. This approach makes the SLA control system more flexible. Below, Table 5 describes some properties of violation data concepts and alarm managements.

Table 5. Violation Data Concepts and Alarm Managements.

Object Property	Domain	Range	Datatype Property
ConcernService	AlarmViolationSlaM.	Service	hasLname
hasAlarmViolationSlaManagementProfil	PatientCopd	AlarmViolationSLA.	hasDate
concernsSLA	AlarmViolationSLA.	SLA	hasSLA_id
hasIncrementAlarmLevel	AlarmViolationSLA.	IncrementAlarmLevel	IncrementAlarmLevel

Figure 8 shows Violation Data Concepts and Alarm Management.

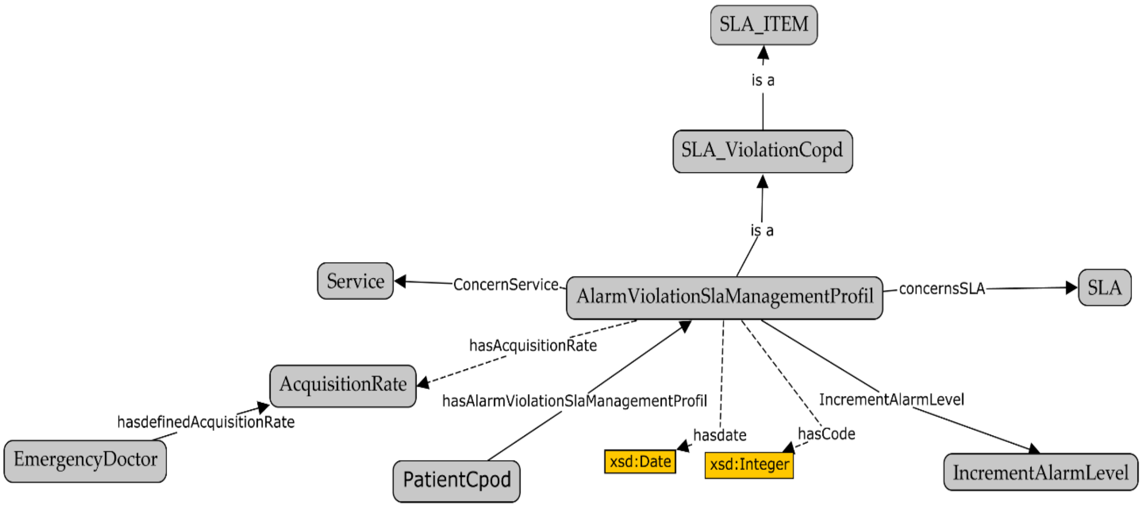


Figure 8. Violation Data Concepts and Alarm Management.

(v) ViolationSLARequestconcepts and Alarmanagement Module

The purpose of this ontology is to record violations informations. Then, it provides the information to the template dynamically. The items that mainly participate in the SLA control mechanisms are ViolationRequest_Concepts and Request. The parameters hasNumbeRequestSent and hasNumberRequest are the data properties. Their values are calculated at the end of each AcquisitionRate collection period with the request incident control algorithm. If there is an incident request, then an alarm is generated with the hasGenerateAlarmRequest relationship. The incident request counter is then incremented with the relation hasIncrementAlarmLevelRequest. Figure 9 describes the different relations between concepts.

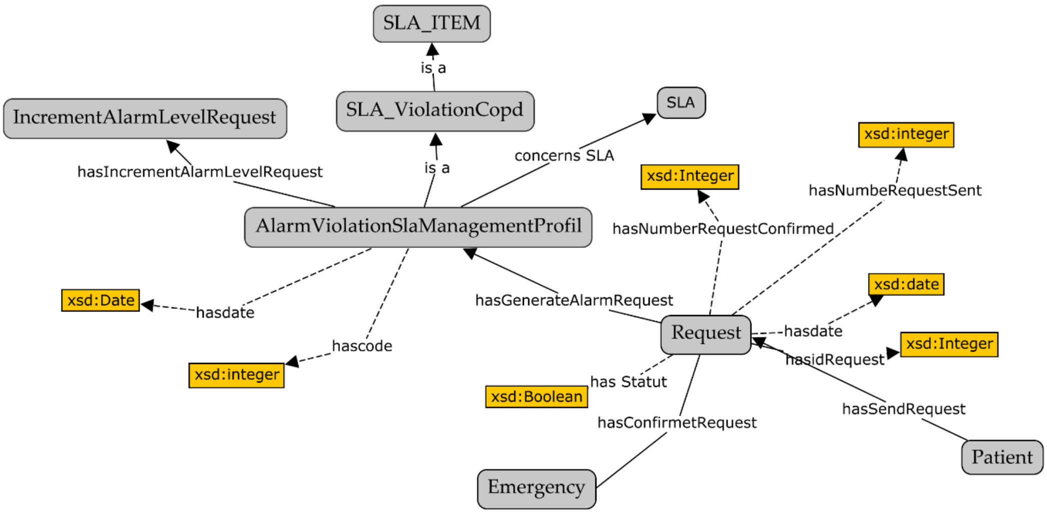


Figure 9. Violation SLA Request concepts and alarm management.

The main branch of the ontology related to SLA control mechanisms is Request, ViolationRequest _Concepts. An incident request occurs if the sent request has not been received by the emergency physician, i.e., the status of the request is not confirmed. The difference between the number of requests sent and the number of confirmed requests in a period determines the number of request incidents. The concept or class allows the recording of these incidents and to populate the virtual template dynamically according to the period.

(vi) Subscription to Service Monitoring COPD Ontology Module

This ontology is not directly involved in the control mechanisms of the SLA. However, it provides important information to enable the control mechanisms to work correctly and above all, to adapt the interfaces to the context. Indeed, it provides information about the connected patient and the services affected by the SLA. The items that mainly participate in the SLA control mechanisms are: patients, services, and SLA actors. The hasSubscribeSystemCpodMonitoring relationship allows the subscription of the COPD patient, the emergency physician and the triage agent. For each contract with the COPD monitoring system, a service or services are assigned to the subscribed patient. Table 6 describes the relations between the concept service components.

Table 6. Some properties of Subscription Ontology.

Object Property	Domain	Range	Datatype Property
hasSubscribeSystemCpodMonitoring	Patient	SubscribeSystemCpodMonitoring	hasLname
hashospitalcenter	AgentTriage	HospitalCenter	hasNameAgent
hasconcerned	SubscribeSystemCpodMonitoring	Service	hascode
hasSubscribeSystemCpodMonitoring	EmergencyPhysician	SubscribeSystemCpodMonitoring	hasEPname

Figure 10 describes subsciption module.

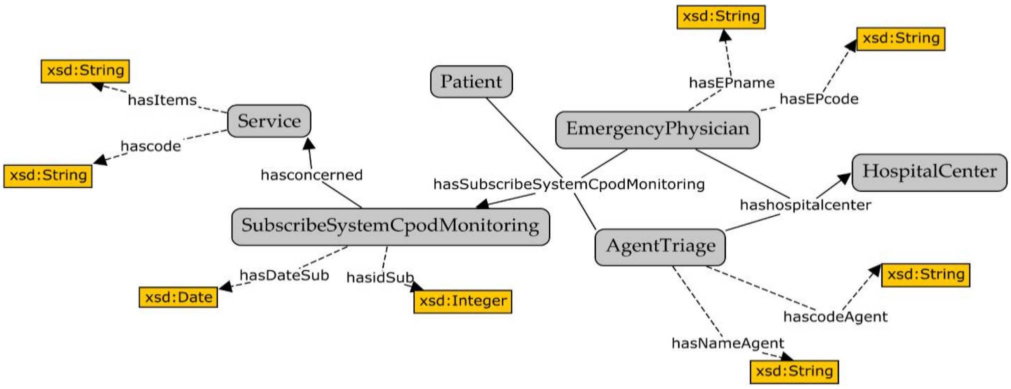


Figure 10. Subscription to Service Monitoring chronic obstructive pulmonary disease (COPD) Ontology.

3.2.4. Acquisition Layer

The collected data refer to the parameters presented in the table below but for the evaluation of the SLA, we focused on vital parameters only. Table 7 describes the information of the patient parameters.

Table 7. Parameters List.

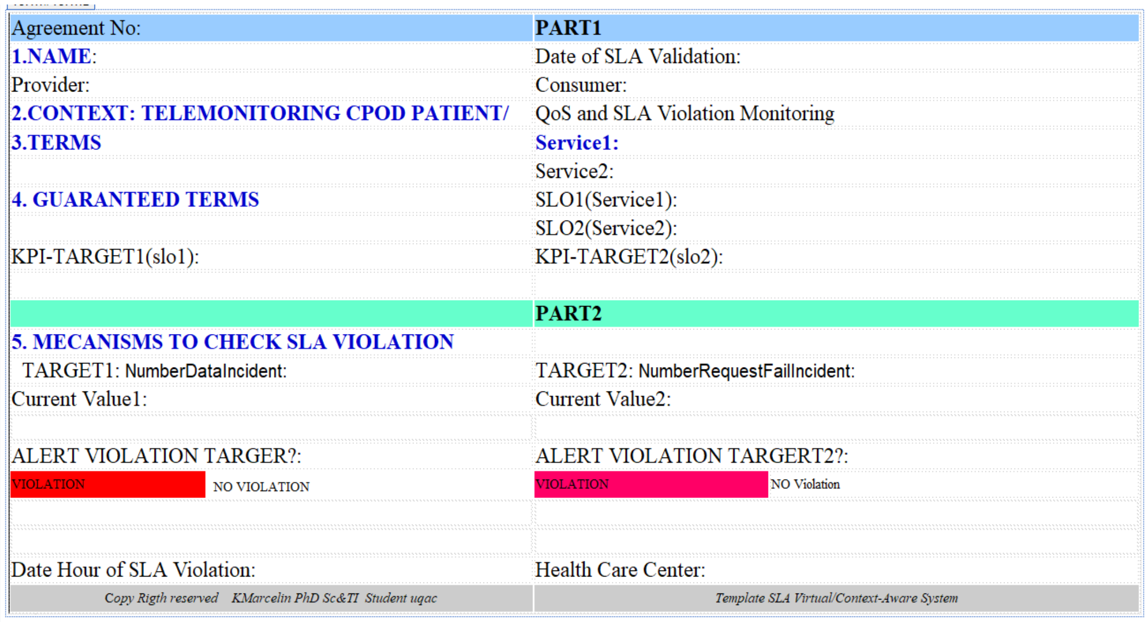
1. Profile Parameters	2. Symptom Parameters	3. Vital Sign Parameters
1. Gender	5. Dyspnea	12. HR (Heart Rate)
2. Age	6. Shortness of Breath	13. Pulse Ox (Spo2)
3. BMI	7. Cough	14. Fev1
4. Medical History	8. Wheezing	15. Temperature
	9. Sputum	
	10. Infection	
	11. Respiratory Symptoms Wake You Up	

There are two categories of data. In first category, there are the static data: age, BMI, gender, and medical history. These data are available in the patient's medical records and only the emergency physician can consult them. In the other category, there are the dynamic data collected from the sensors: vital parameters and symptom parameters. The decisions in a monitoring system are fundamentally based on the quality of the data. Incorrect data can influence the emergency physician's decision-making process. A wrong decision can hurt the patient's health. Evaluating the data aspects of the SLA is intended to improve the QoS. The other aspect of the SLA assessment concerns the requests transmitted to the emergency physician by the patient. An request incident triggered when a request sent by the patient is not received by the emergency physician for any reason.

4. Design of Virtual Adaptive Template SLA and Its Implementation

4.1. Design of Virtual Adaptive Template SLA

The purpose of a virtual SLA is to dynamically monitor the QoS guarantees of the SLA in a COPD patient monitoring context. This template has two parts. Part 1 represents the offered service that has been validated by the parties. If the data of Part 1 is changed, the virtual template will also be modified in real time. In addition, part 2 presents the dynamic control elements of the SLA. This part adapts dynamically according to the data of the patient context and the context-aware ubiquitous system. Finally, the information described in these two parts is extracted from the ontology presented in the previous sections. To allow a better understanding of the virtual SLA template, important concepts are presented. According to reference [17], the SLA template must have important information: SLO, KPI-TARGETs, Condition, and Constraints. Figure 11 describes the virtual SLA template.



Agreement No:	**PART1**
1.NAME:	Date of SLA Validation:
Provider:	Consumer:
2.CONTEXT: TELEMONITORING CPOD PATIENT/	QoS and SLA Violation Monitoring
3.TERMS	**Service1:**
	Service2:
4. GUARANTEED TERMS	SLO1(Service1):
	SLO2(Service2):
KPI-TARGET1(slo1):	KPI-TARGET2(slo2):
	PART2
5. MECANISMS TO CHECK SLA VIOLATION	
TARGET1: NumberDataIncident:	TARGET2: NumberRequestFailIncident:
Current Value1:	Current Value2:
ALERT VIOLATION TARGER?:	ALERT VIOLATION TARGERT2?:
VIOLATION NO VIOLATION	VIOLATION NO Violation
Date Hour of SLA Violation:	Health Care Center:
Copy Rigth reserved KMarcelin PhD Sc&TI Student uqac	*Template SLA Virtual/Context-Aware System*

Figure 11. Virtual SLA template.

Concerning the SLO, Labidi et al. [17] explained: "*The Service Level Objective element in a guarantee term is also expressed as an assertion over service attributes and/or external factors such as date, time. However, most often a Service Level Objective is expressed as a target for a Key Performance Indicator (KPI) such as average response time, completion time*". Concerning KPI-Targets, Labidi et al. [17] concluded: "*A KPI Target expresses a service level objective by specifying a target for a key performance indicator (KPI) such as average response time, availability, etc. associated with a service. The definition of a KPI is outside the scope of the current specification*".

4.2. Implementation

We used the Ontopy platform to implement our prototype. According to [14] Ontopy is a free software Python module (GNU LGPL v3 license) for ontology-oriented programming and agile ontology-based application development. The Python 3.4 language was chosen because it is a dynamic object language with multiple inheritance.

Figure 12 describes the architecture of the Ontopy platform.

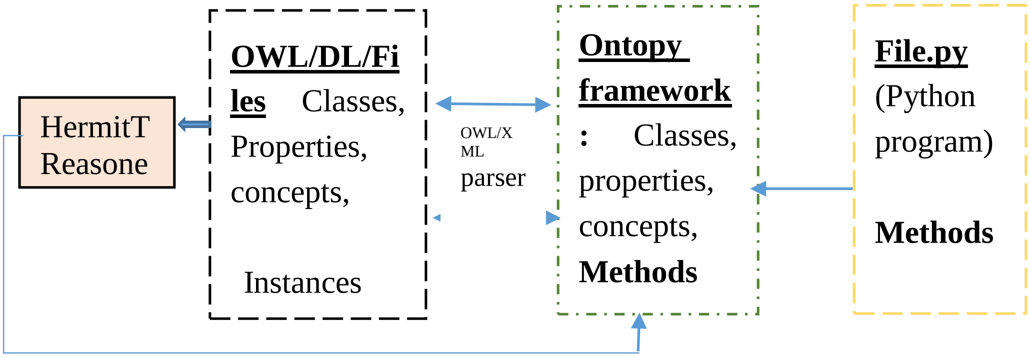
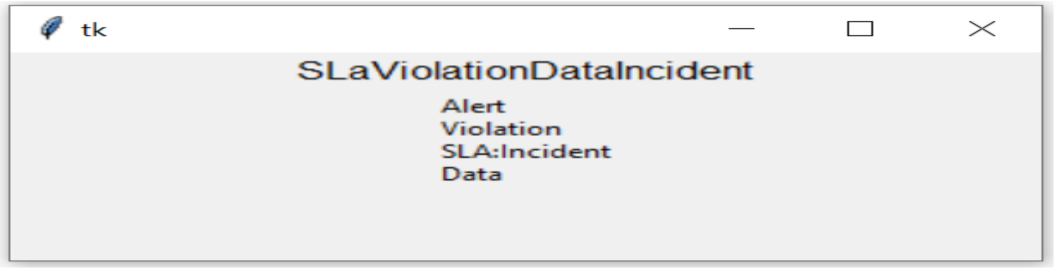

HermitT Reasone
OWL/DL/Files Classes, Properties, concepts,
Instances
OWL/XML parser
Ontopy framework: Classes, properties, concepts, Methods
File.py (Python program)
Methods

Figure 12. Architecture of Ontopy Platform [14].

We designed the ontology with Protégé, a tool for Ontology Web Language, Decidable Language(*OWL/DL)* projects. We exported classes, instances, and relations into the Ontopy platform. We used Python 3.4 to implement a simple application prototype. Our implementation scenario checks the feasibility of the SLA evaluation virtual model. To this end, we defined three incidents of data and two incidents of service request acceptable per month as the SLA KPI_TARGET value. We simulated the data for the patient context with four files containing the temperature parameter. The temperature values in the files f1, f2, f3, and f4 are erroneous or aberrant values. The structure of the files is f (idpatient, temperature): f1(1,20), f2(2,55), f3(3,60), and f4(4,100). The program reads these files and calculates the number of incidents, which is 4, and checks that 4(IncidentDataNumber) > 3(KPI_TARGET value of the SLA). Then, the prototype triggers the SLA violation alert and the virtual template publishes all the SLA violation information. Figure 13 describes the results of this test with Python 3.4.



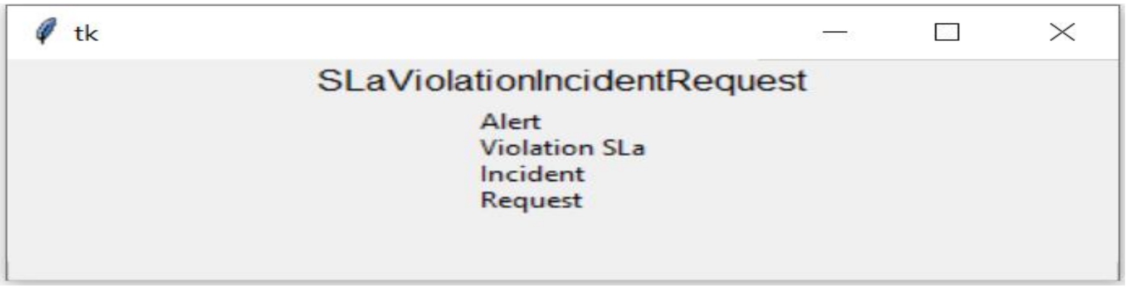
SLaViolationDataIncident

Alert
Violation
SLA:Incident
Data

Figure 13. Violation incident data test.

For request incidents, we created two files, A1 and A2. Both files contain, respectively, the number of requests sent by the COPD patient to the emergency physician and the number of requests received by the emergency physician. The structure of the files is A (id, NumberRequest): A1(1,6); A2(2,3). The program reads the files and calculates the difference between the sent and received requests. Then the program compares this result to KPI-TARGET value extracted from the SLA. The system triggers the SLA violation alert and the dynamic model publishes these results.

The Figure 14 presents a simple example of an incident request test.



SLaViolationIncidentRequest

Alert
Violation SLa
Incident
Request

Figure 14. Incident Request Test.

5. Conclusions and Future Works

Several contributions of this study can be noted. One of the major contributions is that, however large or small the volume of data manipulated by the COPD monitoring platforms, our SLA violation control mechanism allows us to monitor the quality of the data. Firstly, our solution realizes the dynamic evaluation of the SLA by extracting relevant data from the SLA documents and extracting dynamic data from the COPD patient to populate an SLA violation control ontology. Secondly, our solution provides virtual and adaptive SLA template for SLA violation prevention. Thirdly, virtual SLA template is accessible to all SLA parties from their context-aware application interface. Finally, the implementation of the SLA approach to better control quality of medical monitoring platform service is a relevant step for Health Care System and Health Information Technology (HIT) research area. In our future research projects, we will investigate if patient data collection periods influence the SLA violation.

Author Contributions: K.-M.K. designed the presented model. K.-M.K., H.M., and H.A. analyzed the computational framework of the ontology. K.-M.K. wrote the manuscript in consultation with H.M. and H.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was sponsored by Natural Sciences and Engineering Research Council of Canada (NSERC), and the computer science department of the University of Quebec at Chicoutimi (Quebec), Canada.

Acknowledgments: All my thanks to Hicham Ajami in TI of the Department of Computer Science and Mathematics at UQAC.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, Z.A.; Sivakumar, S.; Phillips, W.; Robertson, B. A QoS-aware routing protocol for reliability sensitive data in hospital body area networks. *Procedia Comput. Sci.* **2013**, *19*, 171–179. [CrossRef]
2. Tamura, G. QoS-CARE: A Reliable System for Preserving QoS Contracts through Dynamic Reconfiguration. Ph.D. Thesis, ICESI University, Cali, Colombia, 2012.
3. Abawajy, J.; Fuzee, M.F.; Hassan, M.M.; Alrubaian, M. Service level agreement management framework for utility-oriented computing platforms. *J. Supercomput.* **2015**, *71*, 4287–4303. [CrossRef]
4. Khattak, A.M.; Akbar, N.; Aazam, M.; Ali, T.; Khan, A.M.; Jeon, S.; Lee, S. Context representation and fusion: Advancements and opportunities. *Sensors* **2014**, *14*, 9628–9668. [CrossRef] [PubMed]
5. Shah, T.; Yavari, A.; Mitra, K.; Saguna, S.; Jayaraman, P.P.; Rabhi, F.; Ranjan, R. Remote health care cyber-physical system: Quality of service (QoS) challenges and opportunities. *IET Cyber Phys. Syst. Theory Appl.* **2016**, *1*, 40–48. [CrossRef]
6. Kouame, K.M.; Mcheick, H. Architectural QoS pattern to guarantee the expected quality of services at runtime for context-aware adaptation application. *SN Appl. Sci.* **2019**, *1*, 405. [CrossRef]
7. Comuzzi, M.; Marquez-Chamorro, A.E.; Resin, M. A Hybrid Reliability Metric for SLA Predictive Monitoring. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, Limassol, Cyprus, 8–12 April 2019; pp. 32–39.
8. Sahoo, P.K.; Mohapatra, S.K.; Wu, S.-L. SLA based healthcare big data analysis and computing in cloud network. *J. Parallel Distrib. Comput.* **2018**, *119*, 121–135. [CrossRef]

9. Omar, W.M.; Taleb-Bendiab, A. Defining an Ontology for E-Health Autonomic Software Services. *Innov. Inf. Technol. Dubai* **2006**, 1–5. [CrossRef]
10. Widmer, T.; Premm, M.; Scheele, M.; Murray, J.; Karaoke, P. Distributed Knowledge Sharing for Patient Guidance Ehealth Services. In Proceedings of the ECIS 2013: 21st European Conference on Information Systems, Utrecht, The Netherlands, 6–8 June 2013; p. 199.
11. Benlamri, R.; Docksteader, L. MORF: A mobile health-monitoring platform. *IT Prof.* **2010**, *12*, 18–25. [CrossRef]
12. Labidi, T.; Mtibaa, A.; Gargouri, F. Cloud SLA terms analysis based on ontology. *Procedia Comput. Sci.* **2018**, *126*, 292–301. [CrossRef]
13. Andrieux, A.; Czajkowski, K.; Dan, A.; Keahey, K.; Ludwig, H.; Nakata, T.; Pruyne, J.; Rofrano, J.; Tuecke, S.; Xu, M. Web Services Agreement Specification (WS-Agreement). In Proceedings of the Open Grid Forum, Seattle, Washington, DC, USA, 15–19 October 2007; Volume 128, p. 216.
14. Kritikos, K.; Plexousakis, D. Semantic QoS Metric Matching. In Proceedings of the 2006 European Conference on Web Services (ECOWS'06), Zurich, Switzerland, 4–6 December 2006; pp. 265–274. [CrossRef]
15. Lacot, X. *Introduction à OWL, un Langage XML d'Ontologies Web*; École Nationale Supérieure des Télécommunications: Paris, France, 2005.
16. Ajami, H.; Mcheick, H. Ontology-based model to support ubiquitous healthcare systems for COPD patients. *Electronics* **2018**, *7*, 371. [CrossRef]
17. Fakhfakh, K.; Chaari, T.; Tazi, S.; Drira, K.; Jmaiel, M. A comprehensive ontology-based approach for SLA obligations monitoring. In Proceedings of the 2008 Second International Conference on Advanced Engineering Computing and Applications in Sciences, Valencia, Spain, 29 September–4 October 2008; pp. 217–222.