

Context Description and Next Activities

The figure 1 illustrates our data integration scenario. Cloud providers (for instance, Cloud Provider A, Cloud Provider B and Cloud Provider C) offer cloud resources to data providers (for instance, Data provider 1, Data provider 2, Data provider 3 and Data provider 4) willing to deploy their services. The cloud provider and the data service establish a contract specifying what guarantees in terms of infrastructure resources (for instance, storage limit, memory limit, processing capacity) the data service can expect from the cloud. This contract is called *Cloud SLA* (SLA_C).

Data services can deploy services in the clouds they have subscriptions respecting what is agreed in the SLA_C . Each service deployed by the data service in the cloud export a different SLA (called *Service SLA* - SLA_S) which specifies what service customers can expect in terms of data quality guarantees (for instance, provenance, freshness, data type, degree of rawness) from its service. The data provider defines the SLA_S for the services deployed on a cloud according to what it is defined in the SLA_C . For instance, considering that a *data provider* have agreed with a *cloud provider* to have limit of 10 gigabytes of free data transferred per day, the *data provider* could define on his SLA_S that he can perform 300 requests per day, and each request costs 0.1\$. In other words, the SLA_S guarantees are derived from the SLA_C . Moreover, a *data provider* can deploy the same service in different clouds in which he has established contracts (for instance, the *Data provider* 1 deployed the service S1 in the clouds A and B) and for each different *Cloud provider* a different SLA_S is defined for the same service.

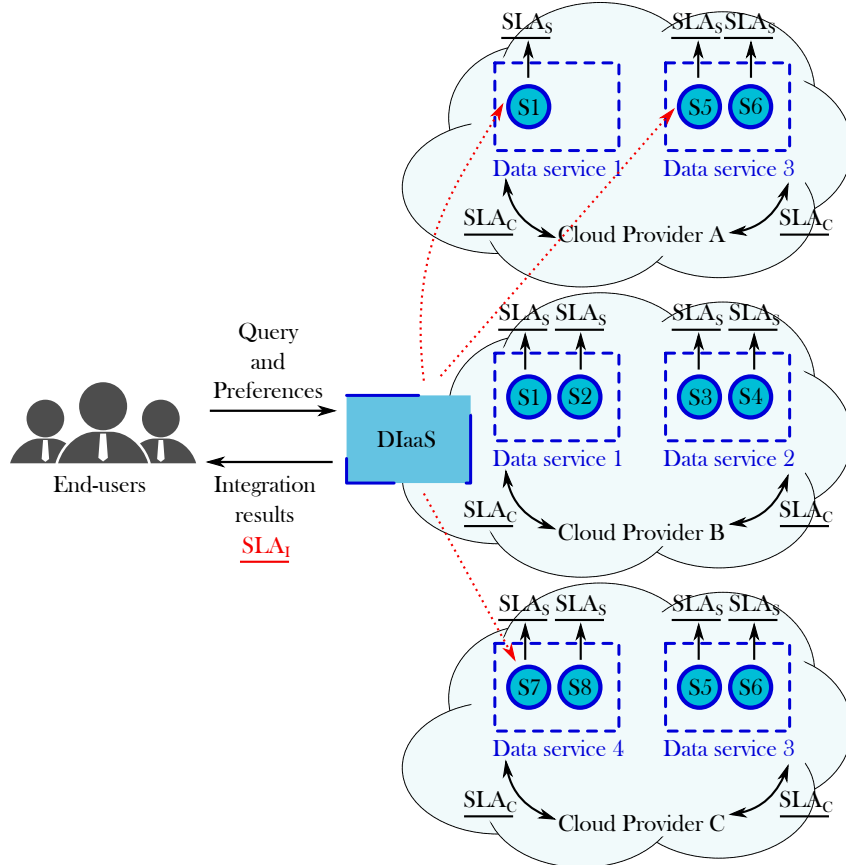


Figure 1: Data integration context overview

The end-user willing to integrate data interacts with our *Data Integration-as-a-Service (DIaaS)*. The *DIaaS* is responsible to select and match the services that can produce the result expected by the user according to his preferences, where he is consuming the data, and the different SLAs associated to the services and to the cloud providers. Once the composition is created and executed, the integration results are delivered to the user and an *integration SLA* (SLA_I) is established. This SLA is responsible to include information collected during the integration process which can be reused in a further integration request.

Activities for the next meeting:

1. Enumerate the possible cases of queries and their variations to be able to identify what could be of interest to be in the *integration SLA* and what information should I collect in the integration process.
2. Propose the schemas of SLA_C and SLA_S according to the schema that I have removing what is not necessary for it (clauses extraction from SLA and how those clauses will feed the rewriting algorithm: study all the cases: clauses are compatibles, clauses are partially compatible...etc).

List of query variations

Query	Requirements
The incoming query is the same as a previous query	Same requirements
	Requirements more restrict
	Requirements less restrict
	Part of the requirements more restrict and part of the requirements less restrict
	Part of the requirements more restrict and part of the requirements different
	Part of the requirements less restrict and part of the requirements different
	Requirements completely different
The incoming query is a subset of a previous query	Same requirements
	Requirements more restrict
	Requirements less restrict
	Part of the requirements more restrict and part of the requirements less restrict
	Part of the requirements more restrict and part of the requirements different
	Part of the requirements less restrict and part of the requirements different
	Requirements completely different
The previous query is a subset of the incoming query	Same requirements
	Requirements more restrict
	Requirements less restrict
	Part of the requirements more restrict and part of the requirements less restrict

Query	Requirements
	Part of the requirements more restrict and part of the requirements different
	Part of the requirements less restrict and part of the requirements different
	Requirements completely different
Different queries but the incoming query has some abstract services in common with the previous query	Same requirements
	Requirements more restrict
	Requirements less restrict
	Part of the requirements more restrict and part of the requirements less restrict
	Part of the requirements more restrict and part of the requirements different
	Part of the requirements less restrict and part of the requirements different
	Requirements completely different
Different queries	Same requirements
	Requirements more restrict
	Requirements less restrict
	Part of the requirements more restrict and part of the requirements less restrict
	Part of the requirements more restrict and part of the requirements different
	Part of the requirements less restrict and part of the requirements different
	Requirements completely different

Table 1: List of possible query variations

Cloud SLA (Figure 2)

Service SLA (Figure 3)

Integration SLA

By considering the list of query variations, the following clauses are interesting to be part of the *integration SLA* in order to optimize a further integration request.

1. *The complete query definition.* The query is necessary to identify similarities with a further integration request.
2. *The list of user preferences.* The *user preferences* are necessary to identify similarities with a further integration request.
3. *Integration total cost.* The integration final cost could be a *user preferences* being used a filtering measure.

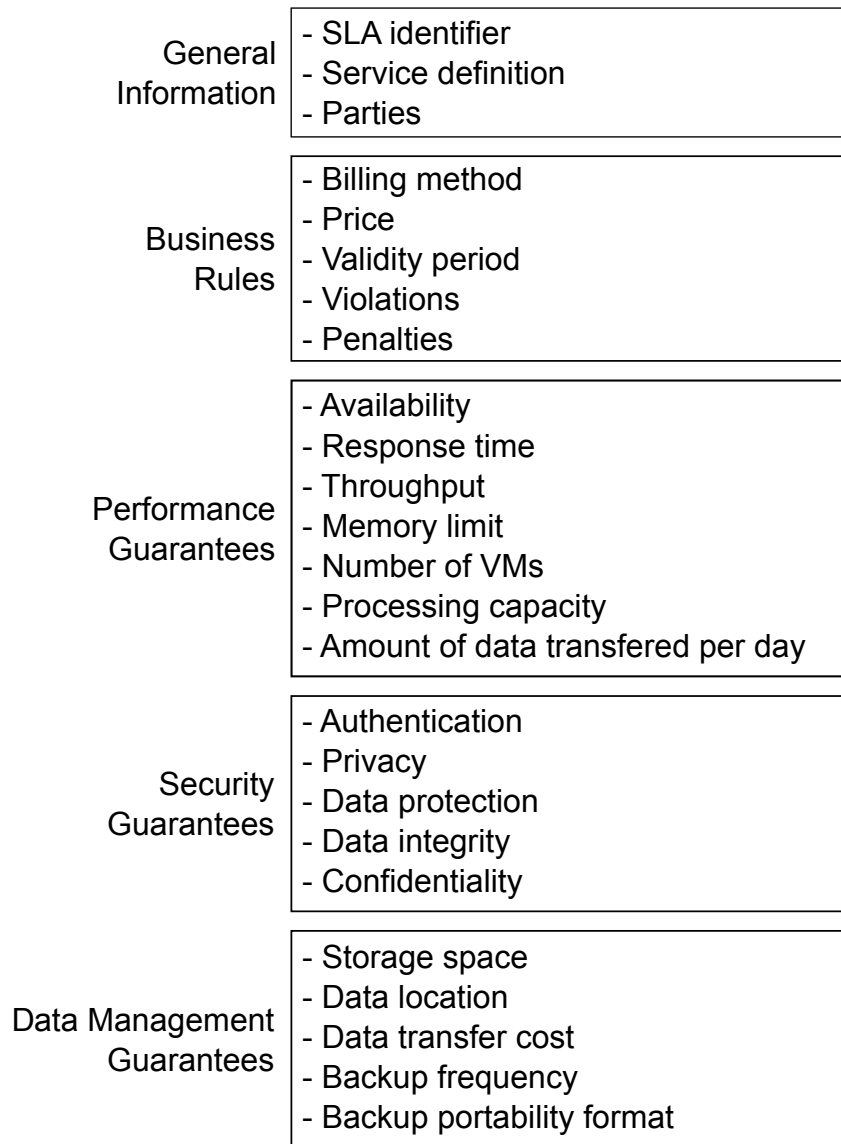


Figure 2: Cloud SLA schema

4. *Integration time.* The necessary time to perform the integration process. This could be a filtering measure considering the *user preferences*.
5. *Used data services.* The *data services* that were used in a previous integration process.
6. *User data consumption environment.* The environment that the user is using to consume the data.
7. *Amount of data transferred.* The amount of data that were transferred and delivered to the user.
8. *Consumption time.* The time necessary to deliver the results to the user considering his consumption environment.

The following information seems to be interest for the integration process. However, it is not an information to be included in the *integration SLA* once it is something general

General Information	<ul style="list-style-type: none"> - SLA identifier - Service definition - Parent SLA identifier
Business Rules	<ul style="list-style-type: none"> - Validity period - Price per call - Number of request per day
Performance Guarantees	<ul style="list-style-type: none"> - Availability - Response time
Security Guarantees	<ul style="list-style-type: none"> - Authentication - Privacy - Confidentiality
Data Quality Guarantees	<ul style="list-style-type: none"> - Data type - Degree of rawness - Veracity - Production rate - Production time - Provenance - Freshness - Trust

Figure 3: Service SLA schema

for every integration request. I believe it should be included in a file apart.

Abstract services description (I do not know if it is the best name for it): it is a file that associates an *abstract service* to *data services* that can answer to it including performance information of the *data service*.

This file should include the information regarding the *data services* that can cover an *abstract service*. In the case when no reusable integration exists, this file would avoid the effort of searching for *data services* that can cover an *abstract service* when a new integration request arrives. Even in the case when a reusable integration exists, it could help while identifying and substituting *data services* that are not good in performance or that are not interest considering the current *user preferences*, for instance. Moreover, during the integration process information concerning the processing time of each involved *data service* is collected and included in this file. This information could be interest while choosing a *data services* for a further integration process.