ABSTRACT
Currently several enterprises depend on services provided to them by other parties for the realization of their own service offer. The interconnections in terms of services offered and required by companies shape complex systems, called service networks. The new cloud-computing paradigm is stressing the concept of the governance of relationship between IT customers and providers. The relationship is based on Service Level Agreements (SLA) and obligations, described inside contracts between providers and customers. Every customer needs to agree with a SLA in order to lease a new service. SLAs describe provisioning terms and encapsulate QoS characteristics as well as functional properties. Traditionally, providers define SLAs in which they guarantee explicit provisioning service level bounds for an agreed period. In the scientific literature, SLAs are hardly viewed as end-user documents, and merely as automated processes that assist the monitoring and scheduling of resources. In contrast, cloud IT marketplaces treat SLAs as static documents that do not allow for any processing. Moreover, strong diversity exists in how service providers coming from distinct business and socio-economical domains formulate and exercise their provisioning responsibilities. Such issues need to be resolved in order to make SLAs enacting tools for managers and services’ providers in order to govern contracts. The contribution of this paper is to discuss some relevant open issues not covered by current contract management approaches and tools. In particular we will explore how service composition, with specific focus on service levels, limits and shapes technical, legal and organizational aspects related to the contract of the composed service.

Categories and Subject Descriptors

General Terms
Performance; Measurement; Management; Reliability

Keywords
SLA Management, contract management; cloud computing; Business Process Outsourcing

1. INTRODUCTION
Services are becoming a pervasive element in the current economic scenery. Several companies and professionals rely on the consumption and provision of services to ensure their processes and to realize their specified deals. Currently, companies all over the world are interconnected with each other by complex service-centric grids called service networks. The ubiquity and pervasiveness of service networks create the need for models, methods and instruments to understand them and join their potential.

Business Process Outsourcing (BPO) and cloud computing are some examples of the recent acceleration in the evolution of the Internet as technological vehicle underpinning the expansion of service provision and inter-/intra-enterprise integration in all market sectors. These paradigms push to a new vision of IT management, not limited to technology and more oriented towards final users. The delivery of IT services (especially software as a service) to final customers requires the description of the service in end-to-end perspective, which means that IT manager should be able to map and monitor the whole IT service chain from the data center to the final user. Actually, in scenarios like BPO or cloud computing, the IT manager must monitor the service levels as agreed in the signed contract with his service provider.

Everyone taps into services every day by using social networks, online banking facilities, web mails, desktop sharing tools, groupware applications, etc. However, services are not only an Internet-related occurrence. For example, education at large can be seen as a service system, as well as call centers, IT support [29], and telecommunication [30]. From the business perspective, there are different paradigms for creating enterprise architecture. The most important is to encapsulate the functionalities of IT resources as services. By this means, it is possible to clearly describe the contributions of IT both in terms of functionality and quality and to define a service-oriented enterprise architecture (SoEA). SoEA easily integrates wide-spread technological approaches such as Service Oriented Architecture (SOA) or emerging ones as cloud computing, because they also use service as structuring and governing paradigm.
In these scenarios, the IT manager usually signs several contracts with service providers based on a proper trade-off between cost and quality of service [31]. Then the IT department owns the task to compose these services in order to generate new services to the final user.

Business Process Outsourcing and cloud computing paradigms, representing the current paramount examples of service networks, are laying emphasis on the contract as a key element to govern the relationship between IT customers and their providers. In general, these documents (the contracts) include legal aspects, terms, obligations, Service Level Agreements (SLA), QoS (Quality of Service) and all relevant functional and non-functional properties of the service. Therefore, contracts are central not only to govern the operational aspects of the service, which is the subject of the contract, but also to define economic and temporal features of the relationship among the involved parties.

This is one of the reasons why contracts for online services, and specifically for IT services, are rapidly evolving from their shape of static documents, signed by humans and enforced by law, to electronic documents orchestrating the interplay of human and automatic agents belonging to networked organizations.

This new scenario is challenging for contract owners, because contracts, or part of them, must be readable and meaningful for both human and automatic agents. Therefore, a comprehensive and systematic evaluation of different aspects (e.g. technical, organizational, legal, economic, financial) [2] is required, in order to support contract owners to fully understand the interwoven effects of their choices and to minimize possible risks.

Moreover, in networked organizations different services, deployed by multiple providers, can be composed to provide a new service, whose contract is subjected to constraints coming from the underpinning contracts. In this case, the contract owner of the composed service must also evaluate inter-organizational aspects of contract clauses to assess potential incompatibilities, risks and restrictions. For example if two services are available in disjoint temporal windows, they cannot be cascaded because the resulting availability window would be empty. The same would happen if the "privacy" clauses, or other legal aspects related to the contract of the component services, are incompatible.

In this sense, SOAs (Service Oriented Architectures) and service composition techniques are necessary but not sufficient to "contract composition". In effect service composition field, well known in literature for the functional part, doesn’t cover non-functional and non-technical elements included in contracts for online services. To deal with these problems a systematic approach and a suitable tooset is needed to support contract composition. The contribution of this paper is to discuss some relevant open issues not covered by current Contract Management (henceforth, CM) approaches and tools. In particular we will explore the feasibility of service composition and, with a specific focus on service levels, how it limits and shapes technical, legal and organizational aspects related to the contract of the composed service.

The paper is structured as follows: section 2 describes the research background and the related works, section 3 reports the main issues related to composition specifications of services with focus on the impacts of service level features on the composed service contract. Section 4 reports how different elements or variables impact on service composition and related contract. Section 5 is for conclusions and further works.

2. BACKGROUND AND RELATED WORKS

2.1 Our experience

For 5 years we have been studying a large company which operates in the mobility sector (henceforth named MobEnt), supporting the internal IT department in its transformation from IT resources provider to IT service provider. In this process MobEnt has outsourced the whole IT data center and its software applications to an external provider through a seven years contract. MobEnt counts more than 300 applications developed in 30 years, covering a wide variety of architectures, technologies and domains.

The new outsourcing contract has defined SLAs for the availability and performance (seen as response time) of the application service, pushing the IT organization (some hundreds of employees) to change their approach from IT hardware and software managers to IT service managers. Moreover, IT service managers have been more and more involved in the definition of business performances related to online business services (e.g. availability of eticketing service, number of sold etickets, time of check-out).

Furthermore, all new business services have been conceived and designed as a composition of lower level services provided through different technological platforms. A non-trivial problem regards "how" to define and evaluate the target service levels (and, accordingly, SLAs and related contracts) related to component services, in order to attain a given Service Level (SL) for the business service. IT service managers, in fact, are usually required to meet the SLs requirements to the business in order to better satisfy its customers. A different approach could either be too expensive or not meet customers’ criteria.

A second relevant problem is about the different languages and goals of IT operators and business ones. The problem, already discussed in literature ([28], [11]), was even more emphasized by the fact that the new outsourcing contract has been assigned to a TGC (Temporary Group of Companies) made by firms with different skills, different monitoring standards and different CM procedures.

A consequent problem concerns accountability: when a composed service is based on component services deployed by different operators in the TGC, the related contract should define the corresponding responsibilities, penalties and obligations for each operator. To solve this problem, the TGC should disclose, in part or in whole, its internal details (technical, organizational …) to MobEnt but no tool or methodology helped in defining the disclosure perimeter.

Moreover, usually IT service managers and technical people involved in system monitoring are not involved in evaluating the business impact of incidents, outages or SLAs violation, as well as contract managers and business-service managers are not involved in evaluating the impacts of contractual clauses (SLAs in particular) on the IT infrastructure.
All these topics highlight that currently available CM and SLM (ITIL, Cobit, etc.) methodologies are not sufficient as tools to effectively support large organization in defining and managing contracts for online IT services and for their compositions.

2.2 Related works

The research literature on CM and SLM has two main topics. On the one hand, some authors study service contracts, their modellization and visual notation. In particular, the modellization, which is useful for the model checking, in order to enact the contract, is mainly performed using a deontic logic. On the other hand, some authors focus their attention on contract lifecycle process. However, these works take into account only automated agents in SOAs.

2.2.1 Service contracts

The research state of art about service contracts is reported in [2], where contract content aspects and contract management are reviewed according to some dimensions. Specifically, Duan focuses on the dimensions of Quality of data (QoD), Quality of service (QoS), Legal issues, Context, Business terms to compare the contract content, and the dimensions of Description of contract, Monitor/Control, Selection, Matchmaking, Composition are introduced to evaluate service contract management. It is interesting to notice that the survey is based on the definition of the service contract lifecycle with a specific attention to the integration of IT operations, business process and economic goals in the contest of the public juridical infrastructure.

Moreover, Comeiro et al. [1] focus their work on the design of business services. They propose the SeCo2 Framework to deal with the heterogeneity of service contracts and evaluate the compatibility between them, because it is propaedeutic to enable the service composition. The framework allows comparing contracts defined with different specification languages (e.g. ODRL-S, WSLA, WSOL). They implicitly assume that an organization can choose to buy technical services from different providers in an electronic market. However, there might be some constraints in the Service Provider selection, for example caused by the duration of the contract, withdrawal penalties or specific predefined contract frames.

2.2.2 Contract modeling

In order to enforce service contracts, different contract modeling approaches have been proposed. In particular, Zou et al. [13] propose a model based on OWL-DL and SWRL to disclose obligations of both e-service providers and consumers. The purpose is to allow service participants to monitor the service contract execution and keep track of obligation fulfillment for each party during service delivery. In this case, a graphical notation based on Colored Petri Nets (CPN) is adopted to formally model contract obligations and their interdependencies.

A very rich stream of literature on contract modeling and checking is based on deontic logic [14]. In [7], Marjanovic and Milosevic presented their initial ideas for formal modeling of e-contracts based on deontic constraints and verification of deontic consistency, including temporal constraints. This approach has been later investigated and enriched. For example, in [15] Governatori et al. go a step further providing a mechanism to check whether business processes are compliant with business contracts. They introduce the FCL logic to reason about the contracts, being based again on deontic logic.

Furthermore, the works by Lomuscio et al. provide another methodology to check whether service compositions are compliant with e-contracts. In [16] they present an approach using WS-BPEL to specify both, all the possible behaviors of each service and the contractually correct behaviors, translating these specifications into automata supported by the MCMAS model checker to automatically verify the behaviors. Though, in [17] they consider a service composition in OWL-S and check with MCMAS if the composition fulfills some contract properties written in a formal language based on epistemic and deontic logic. In both works, the specification of real-time constraints is not allowed because they are not supported by MCMAS and the deontic norms are restricted to obligations.

2.2.3 Visual notation

There are several works defining a meta-model for the specification of e-contracts with the purpose of enacting or enforcing them, through visual notation. In [18] Chiu et al. present a meta-model for e-contract templates written in UML, where a template consists of a set of contract clauses of three different types: obligations, permissions and prohibitions. These clauses are later mapped into ECA (Event, Conditions, Actions) rules for contract enforcement purposes, but the templates do not include any kind of reparation or recovery associated to the clauses. In [19] Krishna et al. proposed another meta-model based on Entity-relationship diagrams that they use to generate workflows supporting e-contract enactment. This meta-model includes clauses, activities, parties and the possibility of specifying the exception handling behavior. This approach is not based on deontic logic and says nothing about including real-time aspects natively. Another approach can be found in [20], where Rouached et al. propose a contract layered model for modeling and monitoring e-contracts. It consists of a business entities layer, a business actions layer, and a business rules layer. These three layers specify the parties, the actions and the clauses of the contract respectively, including the conditions under which these clauses are executed. However, real-time restrictions are not included and the specification of the clauses follows an operational approach. Finally, in [21] Heckel and Lohmann propose to visualize contracts by graph transformation rules over UML data models. This approach is specifically focused on testing.

Contract Oriented Diagrams (C-O Diagram) are also used to visually represent normative texts and electronic contracts, made up of obligations, permissions and prohibitions ([6], [9]), as well as penalties, in case of not fulfillment of their obligations and prohibitions, and absolute and relative timing constrains. The visual model proposed by Martínez et al. consists of a box, which indicates the name of the clause, the agent, normative aspects, reparations, conditions and time restrictions. However, C-O Diagrams, as well as the other formal and visual models, do not distinguish between legal aspects and technical aspects and do not investigate the reciprocal relationships. Moreover, they do not consider the problems coming from service composition for reselling (when a customer buys different services from different providers to resell a composed service based on it). The first drawback could be overcome. For example, the approach
described in [22] could be extended for the checking of contractual compliance of business interactions based on business rules, because Molina-Jimenez et al. represent contract clauses using business rules and describe what events need to be captured from the underlying messaging middleware and how they can be processed in a careful manner to evaluate contractual compliance. The latter drawback must consider the compatibility of contracts.

2.2.4 Contract lifecycle
As regards the second key topic, in [4] contract lifecycle is described from a process oriented multi-stakeholder perspective, showing the interaction of different stakeholders in the definition phase, execution phase and conclusion phase. This model generalizes the one proposed in [3] for Virtual Organizations, where providers must join an Enterprise Network (EN) before the collaboration starts. Indeed, Arenas et al. [3] compose a contract in a Virtual organization (VO) binding legal and technical aspects during the contract preparation phase in a General Virtual Organization Agreement (GVOA). So, technical specifications for service configuration and monitoring are appended to the operational GVOA as machine-readable XML policies to facilitate the self-management of VO operations.

However, due to the heterogeneity of possible context and domains, the description of a unique model of contract lifecycle is very difficult to generalize. For example, Chiu et al. [5] consider both pre-contract drafting and post-contract drafting activities. After the exchange of business information, the contract negotiation takes place and technical and legal clauses are defined. When the contract is enacted, the contract enforcement is carried out, with the sub-activities of monitoring and control.

These models are usually verified, but, for example, in telco’s consumer domain, where a big provider imposes a specific contract to thousands of consumer or cloud computing contracts in SaaS or IaaS, where contract is unilaterally defined by the provider and it can be only accepted by the consumer, without any negotiation.

2.2.5 Service composition
Other issues analyzed in literature are related to the inter-organizational scenario in which service and contract composition occurs. In this case, in fact, it is necessary to define check points in which to measure QoS [10][11] and, moreover, to understand the outcome of the service composition.

Indeed, modeling and monitoring service chains and service networks in business ecosystems is a hot topic in Service Computing and Service Oriented Architecture [23][24][25][26]. Usually, contributions are based on graph models or BPMN models which can be easily translated in BPEL. These models are very effective for technical people and they are not aimed for being used by others than technicians.

However, at the best of our knowledge inside these communities nobody has still investigated how to model service chains taking into account stakeholders perspectives. Starting from our experience and the review of research literature, next section will discuss these issues.

3. CONTRACTUAL ISSUES RELATED TO SERVICE COMPOSITION
ITIL, Cobit and industrial practices do not support SLAs enforcement. The composition of services and the consequent SLA result is a further issue, even less explored. The composed service is again the subject of a contract or of a policy. In order to design systems (i.e. methodologies and tools) to support managers in the drafting of service contracts related to composed services, it is necessary to point out the main issues related to the composition of services with particular attention on service levels.

3.1.1 Service definition and granularity
In order to compose services, it is necessary that they are mapped into a common terminology. Services, which are the subject of a contract, can be defined by different stakeholders within an organization. For example, a manager of a line of business, responsible of the Internet applications in a bank, will talk of services in marketing or business terms and he will ask for service availability or fully availability with completely different terms and meaning from the IT person responsible of the IT operations inside the same organization. IT service can be addressed with different stakeholders perspective (i.e. the line of business, the IT operations, the risk managements, the contract management), and in order to compose services and to evaluate resulting service levels with related constraints, it is necessary to model them at the same level of abstraction. Services can be organized in a service catalog, containing information to bridge the gap and relate the description of a service, as defined by a line of business (i.e. a business service), with the corresponding IT infrastructure and application, the related contract and risk topics. Another aspect related with service definition is accountability, necessary in order to identify who manages the service and related service levels. Accountability imposes that all services deployed (regardless whether by human, machine or software) have the obligation to accept responsibility or to account for their actions [32]. By imposing accountability on services, service clients are provided with the transparency of understanding abnormal behaviors in service collaborations. In service composition the accountability support is important since service processes often rely on external service providers in order to deliver part of the service functionalities. A service system must have effective yet efficient mechanisms to ensure that every external service is delivering a consistent and acceptable level of performance to meet the end-to-end quality of service (QoS) of the whole service process.

Figure 1: Decomposition of a service and the underlying relationship with its elementary services
3.1.2 Service composition topology

Service composition topology refers to the layout of connected services. According to the topology services are composed and service level specifications are calculated differently. To decide the acquisition of a service as part of a final one, the manager needs to know which constraints the final service undergoes. Indeed, this is necessary in order to evaluate the feasibility, the sustainability, the compatibility of the composed service against the requirements. As already discussed in [1] and [12], the service composition topology can be represented as graph of services with four control flow structures: sequence, parallel, condition and loop.

A sequential structure implies that services are executed in sequential order; the parallel structure implies that services belonging to different service chains can be executed in parallel, for example if a service is the back up of another one; the conditional structure means that one service is executed based on a specified condition, while the loop structure means one or more services are executed at specified number of times based on a given condition.

Figure 1 represents the topology of a composed service (Service 11) made up of a hierarchical composition of services, until its atomic ones.

The composed service of the previous example can be also represented as a graph-tree diagram, in order to display the service hierarchy (Figure 2).

![Graph-tree representation of a composed service](image)

3.1.3 Service visibility inside contracts

In networked organizations, all involved parties must cooperate to guarantee the agreed QoS to customers. Therefore, in contracts component services can be described according to one of the following ways:

- as black-box: no internal details are given about the component services, which are solely described in terms of its external parameters (availability, response-time, MTBF ...);
- as white-box: internal aspects of the service are fully exposed to the customer, both at design-time and at run-time, to make them aware about critical aspects, monitoring results, service level violations and possible countermeasures;
- as gray-box: the internal aspect are partially exposed to the customer.

The decision about the adopted way depends, among other strategic and technical aspects, on the critical level of the composed service or on the business relationship between the parties. This decision is essential for contract owner to estimate the associated business risks of the composed service (deterministic approach vs. behavioral one).

3.1.4 Service level metrics and measurement units

Values of service level agreements (SLAs) are based on metrics, which contribute to the assessment of the service’s ability to achieve the desired business objectives. In service composition, resulting service levels depend on composing service levels. The necessary conditions for comparing them are that the metric is the same and the measurement units and compatible (e.g. measures must be related to time, or all measures must be percentages).

3.1.5 Service hours and calendar

In the composition of services the time window (and the service calendar) in which the composed service is guaranteed depends on the composing services. It defines the time and calendar the composed service level is guaranteed. For example a Service A is guaranteed from 8.00 to 20.00 in business days and Service B is guaranteed from 7.00 to 23.00 only in odd days. When Service A and Service B are composed into Service C, the guaranteed windows changes according to composition topology: If \( C = \text{A in parallel with B} \) then \( C \) guaranteed service calendar is the calendar window obtained intersecting A service Calendar and B Service Calendar; if \( C = \text{A in parallel with B} \) and A and B are homogeneous (in the sense that they are the same kind of service), then guaranteed windows is the union of A and B services guaranteed calendars. If A and B are heterogeneous, C service’s guaranteed calendar is the intersection of A and B service hours.

If the composed service has an empty guaranteed window, the Service cannot be guaranteed.

3.1.6 Measurement period

Each component service within the SLA owns a specific measurement period. The measurement period of the composed service is constrained by the corresponding component services.

In general, it is the least common multiple (l.c.m.) of each component’s measurement period, if component services start simultaneously. Otherwise initially the two measurements periods will be out of phase and need to be aligned. For example suppose that A service has a bimonthly measurement period and B service has a three months’ measurement period. The measurement period of C composed service should be the l.c.m. of A and B services, that is 6. Suppose that A service started a month later than B service, then two services will be out of phase and only after a 2 months’ transition period C service would start to be measured with a 6 months’ measurement period (Figure 3).

![Measurement period](image)
3.1.7 Contractual duration
Services are guaranteed in the contractual duration. It means that the contractual duration of the composed service is constrained by the durations of the component services.

3.1.8 Monitoring and monitoring plans
In order to measure the service levels of a composed service, it is required to monitor composing service levels in order to measure the corresponding SLAs and relate them with those of the composed service. It is necessary to develop and implement an appropriate monitoring plan, conducting representative sampling, and to report the results. In order to develop a monitoring plan it is necessary to map the relationships between composing and composed services, as already discussed about the composition topology. The monitoring plan must identify and provide information about each monitoring point. There must be sufficient monitoring points to provide representative sampling of all component service levels. Each monitoring point must be located on the topology and labeled. The label must identify it, using a name or number. This monitoring point label must be used on all monitoring information reporting and it becomes the official identifier of the monitoring point.

The same points will identify monitoring ones both for composing and composed services.

An example about how to choose monitoring points in scenarios like the delivery of application services is described in [11].

3.1.9 Contract terms and conditions
Since contracts are made up of subject, parties, terms and conditions, at least, it is necessary that even the composition of terms and conditions must be related. In the following we discuss some aspects among many others, which we deal with the composition of services and service levels.

3.1.9.1 Context definition
The heading for subsubsections should be in Times New Roman 11-point italic with initial letters capitalized in literature usually authors refer to a model of contract lifecycle ([3][4][5]), but according to different contexts, contract lifecycle can match different patterns and contract contents can vary. Some examples of special environments are agent based mechanism, SaaS and Next Generation Networks, which need to be considered during service contract making and management to cater the new requirement and better profit from them. In cloud service contract a specific aspect to be taken into account is related to jurisdiction(s) in which the provider operates and related details about client’s data ownership and control. For instance, cloud providers can transport and store client’s data in or through foreign countries, so the client becomes subject to laws and regulations she/he may not know anything about. Currently service providers have not done a good job in explaining in which jurisdictions and laws they operate with client’s data and the legal requirements the client must meet. Current contracts about cloud services often lack of statements about data ownership or usage rights, leaving a grey area about what a provider can do or not do with the client’s data, what happens to the client’s data if the service provider is sold to another company and or in case the provider goes out of service. This means that enterprises involved in Cloud Services contracts must deal with complex international contracting for civil and commercial topics.

3.1.9.2 Domain definition
As also drawn in the perspectives of [2], service contracts must be domain specific in order to better fit specific situations of a domain in real world practice. Such service contracts features are usually shared as reusable professional knowledge within a specific domain.

4. FEASIBILITY EVALUATION OF SERVICE LEVEL COMPOSITION
The issues previously discussed concern both the feasibility analysis of service level composition and aspects of service level composition management in service contract enactment.

Regarding the feasibility analysis some issues are blocking for service level composition, since they are preparatory in the definition of the composed service, whilst others must be check but are not blocking since they may produce only empty sets. In Figure 4 the main steps of feasibility analysis and service level composition management are shown.

![Figure 4: Steps of service level composition feasibility analysis and management](image-url)

The steps in the inner dotted lines of the feasibility analysis are preparatory and if they are not completed positively all the other steps cannot be proceeded both in feasibility analysis and in service level composition management. In feasibility analysis all the steps are related to the issues previously discussed. In service level composition management after the definition of the monitoring plan and the check points model, data must be collected and processed according to composition rules derived from the topology and from the other aspects analyzed in the
feasibility analysis. Then reports are generated for composed service levels.

5. CONCLUSIONS AND FUTURE WORKS
The paper deals with the governance of the relationship between IT customers and providers, based on SLAs.

The authors highlight that Service Oriented Architectures and service composition techniques are necessary but not sufficient to “contract composition”, as they do not cover non-functional and non-technical elements included in contracts for online services. The paper mainly explores how service composition, with specific focus on service levels, shapes technical, legal and organizational aspects related to the contract of the composed service and discusses the feasibility of the composition according to different aspects.

In research literature the composition of services and the composition of contracts is largely investigated, but we haven’t found references about the issues of compositing service level agreements, as defined in IT delivery contracts. We have also experienced that these issues are critical in enforcing service delivery contracts.

We concluded that it is necessary to integrate aspects related to service levels design and management in contract management process in order to consider constraints and features related to service composition, which are fundamental in contract enactment and further enforcement.

In this way issues like sustainability and feasibility of service contracts could be partly routed. An approach and tools encapsulating these topics could be very effective for several stakeholders involved in SLM and CM processes and could be useful for what if analysis, assessment for guaranteeing certain service level and sensitivity analysis.

Further works include the development of a platform supporting the composition of service levels, with attention to the constraints they will create at contractual level.

6. ACKNOWLEDGMENTS
This work is partially supported by Engineering Ingegneria Informatio Spa and Essentia. We thank Aurora Bozzardi and Cristian Caione for the helpful discussions.

7. REFERENCES