

# On Elasticity and Constrainedness of Business Services Provisioning<sup>▽</sup>

Lam-Son Lê\*, Hong-Linh Truong<sup>†</sup>, Aditya Ghose\*, Schahram Dustdar<sup>†</sup>

\*School of Computer Science and Software Engineering, University of Wollongong, {l1e, aditya}@uow.edu.au

<sup>†</sup>Distributed Systems Group, Vienna University of Technology, {t.truong, dustdar}@infosys.tuwien.ac.at

**Abstract**—In service-oriented enterprise architecture, provisioning business services is made on top of IT processes, which should be elastic amid the availability of computing resources and the variation of user demand. In addition, the provisioning depends on human resources utilized and is constrained by the business objectives (e.g. a goal) plus coarse-grained constraints (e.g. an order in which business services take place). This elasticity and constrainedness can best be witnessed on non-functional properties of the business services being provisioned. In this paper, we propose a novel framework for modeling and reasoning about them. The framework features a methodology for formally expressing the aforementioned factors in services provisioning, an engine to find solutions and a simulation.

**Keywords**-Elasticity, Non-Functional Properties, Business Processes, Business Objectives, Service-Oriented Enterprise Architecture, Cloud Computing.

## I. INTRODUCTION

Cloud computing increasingly makes impacts on service-oriented enterprise architecture (SoEA). Enterprises no longer solely depend on their in-house enterprise resources and services. Today, resources and services can be sourced from the clouds, off-the-shelf markets, a relatively loose network of people, etc. In SoEA, the concepts of service and resource apply to both business and IT. Apart from typical software-based services, services may be provided by individual humans (henceforth, the term human-provided services) or teams under social computing units [1]. In addition, we have high-level business services that directly support business steps defined in an enterprise specification (i.e. a coarse-grained process that could be as simple as a sequence of purely business transactions). Business services essentially involve (human) service stakeholders some of whom are considered human resources. In a finer-grained description, the business services may rely on underlying IT processes that make direct use of computing resources in the enterprise architecture.

Provisioning business services in this context involves determining non-functional properties (NFPs) that best match a certain pricing (or a rewarding/incentive) model and comply with business objectives (e.g. a goal) and respects business constraints (e.g. ordering specified in business steps). From the perspective of the service providers, as presented in our

notion of *elastic process* [2], service provisioning depends on resources the service providers can utilize. In our work, the resources utilized can be categorized into human resources and software resources, and furthermore, with the emerging cloud computing models resources, can be in-house, owned by the providers or on the cloud, utilized by the provider under pay-per-use models. Therefore, there is a strong correlation between the capabilities of business services and the elasticity of existing resources. This correlation could be interpreted as the *elasticity* of NFPs the providers have advertised for their business services. As an example of the correlation between human resources and the elasticity of service NFPs, the waiting time customers have to spend at a call center to be able to talk to an operator is largely determined by the number of call operators available (simply put, the larger number of operators, the less waiting time). We may also witness the correlation between software-based resources and elastic NFPs. For instance, how to please waiting customers (e.g., while they wait for the next available call operator) is directly attributed to underlying IT capabilities (e.g. chat with a computer-based yet intelligent agent if called from a computer, music broadcasting and vocal messages if called from a phone). In other words, the waiting time and customers' satisfaction (as NFPs) are elastic properties (as opposed to the main functionality which is a static property) of the provisioning of a customer support via telephone (as a service). Also from the service providers' perspective, we need to take into consideration business objectives and steps that *constrain* the dynamic resource requirement and provision.

In this paper, we look at the elasticity and the constrainedness of services provisioning from business down to IT levels. This work leverages our previous work on (i) correlation between business objectives and services [3], [4]; (ii) the elasticity of IT process [2]. More specifically, we propose a methodology where we express business objectives as logic predicates, business constraints as facts, the mechanism to dynamically allocate resources as a set of rules, and the customer demand as parameters. We then devise an engine to find solutions given these factors. We simulate it using a formal language called Alloy [5].

*Paper Structure.* Section II presents the context and motivation of our work. Sections III and IV describe our framework, followed by a prototype in Section V. Section

<sup>▽</sup> Funding of this research was provided by the Smart Services CRC Initiative <http://www.smartservicescrc.com.au/>

VI surveys related work. Section VII concludes the paper and outlines our future work.

## II. BACKGROUND AND MOTIVATION

### A. Emerging Types of IT Services for Business Services

In our work, we focus on the utilization of software-based and human-based resources in the so-called socially-enhanced service computing environments where business services can utilize the following forms of IT resources.

- Software: we consider software that can be provided under the service model.
- Individual human: we consider the capability of an individual that can be utilized as a resource in a business service. In principle, human capability can be provided via the service model, using human-provided service (HPS).
- Social computing unit (SCU): we consider SCUs as teams including different members that perform certain tasks. An SCU can also be provisioned under the service model, in which an SCU can be invoked and paid based on its quality and capabilities.

Conceptually, all these forms of resources can be provided under the service model with well-defined interfaces and NFPs (such as, quality, cost, and availability) [1]. As the number of such services are large in reality, the developers and providers of business services can utilize resources in an elastic manner, thus making their business services being *elastic*. Interestingly, the elasticity is not only bound to software resources but it can also consider human resources.

### B. Modeling Business Objectives

For modeling business objects, we use a simple high-level language for modeling the strategic landscape of an enterprise: the Strategy Modeling Language (SML). The design of the language was guided by the need to provide a mechanism for articulating strategies that would be accessible to senior managers, while still permitting an underlying formal representation (leading to automated tool support).

We view a *strategy model* as a set of *strategy statements* of the following three kinds:

**(1) A goal:** Goals are descriptions of conditions that an organization seeks to achieve. Goals admit boolean evaluation, i.e., an organization is able to clearly determine whether it has achieved a goal or not (consider the following example: “*Our corporate strategy is to be the market leader in mobile handsets*”).

**(2) An objective function:** An objective function is a construct used in operations research techniques to define what the *preferred* or *optimal* solution(s) to an optimization problem might be. These are typically articulated as *maximize  $f$*  or *minimize  $f$* , where  $f$  is a function defined on the *decision variables* (using which the constraints that *feasible solutions* are required to satisfy are also written). Our analysis of a large number of actual corporate strategy

documents, as well as the management literature, suggests that strategies involving *corporate performance measures* or *key performance indicators (KPIs)* are articulated in the form of maximization or minimization objectives. Consider the following statements of strategy: “*Our strategy is to minimize order lead times*”, or, “*Our strategy is to maximize customer satisfaction*”. In the first of these, the intent is to minimize a function encoding order lead time while in the second, a function encoding some definition of customer satisfaction (for instance, using average customer wait times at the customer contact centre, the number of escalations, the number of product returns etc.) is maximized.

**(3) A plan:** A plan is a set of goals together with a set of sequencing and related coordination constraints. In the most general sense, a plan can be as complex as a process model. In this paper, we will view plans only as linear sequences of goals. Also, our analysis of a large number of actual corporate strategy documents suggests that strategies are typically articulated at a very high level of abstraction, where control structures more complex than linear sequencing are never required. A typical example is the following anonymized but actual strategy statement: “*Our strategy is to first gain market acceptance in NZ, then position ourselves in the UK market, then use the UK market credibility to enter the Australian market*”. There are three goals in this strategy, connected via a linear sequencing relation.

### C. Motivating Example

A consulting center offers consulting as a business service via telephone and the Internet. Customers dial a designated phone number describing their needs. Based on this initial description (e.g. consulting domain: finance, banking, family, used cars, dating, etc.), the consulting center will appoint an expert<sup>1</sup> who can best consult them. While waiting for an appointed expert, the customers may enjoy chatting with an operator or computer-based yet intelligent program that is able to answer trivial questions and offer basic recommendation before receiving insightful advices from the appointed expert.

In Figure 1, we model the consulting center using the ArchiMate language in three layers [7]. There are three actors that respectively represent the experts, the operators and the customers. Actors `Expert` and `Operator` refer to human resources of the consulting center. Experts and operators can work together under teams to offer social compute units. Each of these actors plays a specific role<sup>2</sup> (i.e. `Expert` plays `Consultant`, `Operator` – `HelpChat` and `Customer` – `Consultee`). The business service, including two business steps namely `C2P consulting` and

<sup>1</sup>An expert plays similar roles to a call taker at call centers that employ over six million people in North America alone [6].

<sup>2</sup>Role modeling is a mechanism to separate concerns in the early phase of development (e.g. analysis) when a design model does not exist yet.

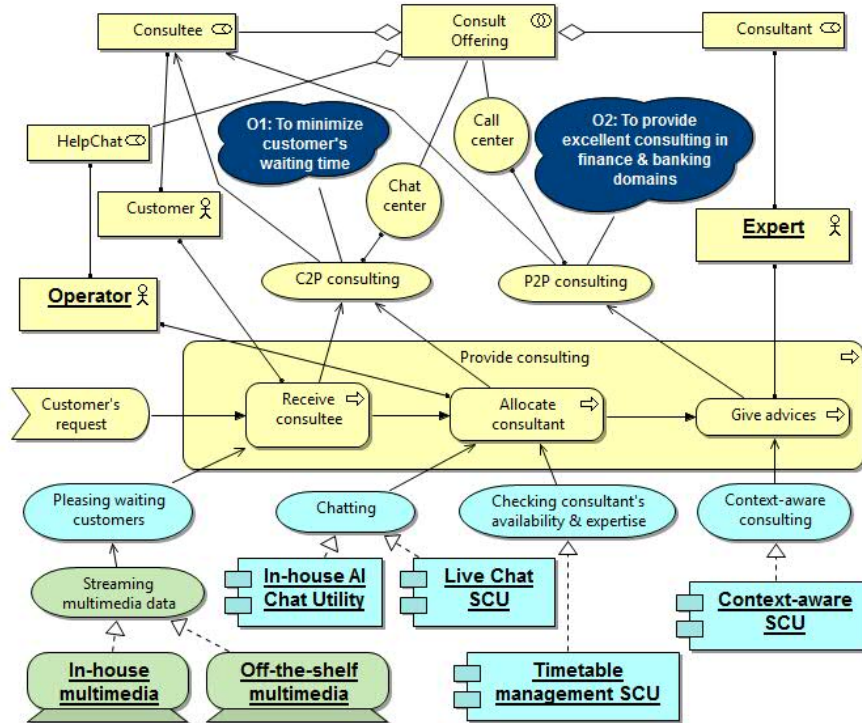


Figure 1. Modeling a consulting center from business down to IT

P2P consulting, provides the customers with computer-based and face-to-face consulting, respectively. The coarse-grained order in which the business service takes place defines business steps, which will be elaborated by IT processes represented in the next layer. The business layer is enhanced by two business objectives that are represented<sup>3</sup> under cloud pictograms. Below these two business steps in the diagram, we have their process view (i.e. process *Provide consulting* and its sub processes).

Further below in the diagram, we have the application and technology layers. They depict resources (software components, humans, middleware and IT services) that are utilized by IT processes implementing the business steps. *Pleasing waiting customers* is a software service, which is used by process *Receive customer*, invokes *Streaming multimedia data* (as a service of the technology layer). This service is realized by *in-house multimedia* and *off-the-shelf multimedia*. *Chatting* is another software service, which is used by process

<sup>3</sup>These two business objectives are abbreviated as *o1* and *o2* together with text describing them in the diagram. In our SML language, *o1* is actually a minimization function whilst *o2* is a goal. We present this enhancement (which is made to the business layer) in light of an extension to Archimate [8] whereby goals are represented under designated pictograms in Archimate diagrams. However the Archi tool did not support this kind of extension by the time we wrote this paper. For this reason, we use the pictogram of Archimate “meaning” to diagrammatically express business objectives in Archi.

*Allocate expert*, requires *in-house chat utility* (with some AI capability) and *live chat* with a technician (as a SCU). *Checking consultant’s availability & expertise* and *Context-aware consulting* are another two software services that (a) help allocate experts based on their availability and expertise; (b) provide the allocated expert with context-aware advices based on customer’s information. They involve two SCUs that deal with timetable management and context-aware computing, respectively.

From customer’s perspectives, this consulting service is utilized via a few NFPs. First, the costs of a consulting section (i.e. an amount of money the customer is charged for using this service) is a combination of a fixed surcharge and a fee that is determined based on a two criteria: period of time for which the consulting session lasts and whether the consulting is conducted at on-business hours or off-business hours. Second, it is necessary to measure the amount of time the customer has to spend chatting with an automatic teller before getting engaged with an appointed consultant. Third, and most sophisticated, it is worth quantifying the accuracy and the insightfulness of advices she received through this consulting service as this property would directly translate to the customer’s satisfaction.

From the service provider’s perspectives, the consulting center maintains a network of consultants whose expertise match the consulting domains they offer to their potential

customers. It also maintains IT resources to realize the software services at the application layer. In Figure 1, human and IT resources are represented under pictograms whose text is highlighted in bold and underline. They are connected to the software services they realize via realization relationships (represented under dashed, triangle-headed arrows). The resources that the service providers can utilize are elastic and in the view of the provider, there are multi-dimensional elastic properties associated with resources, such as cost, availability, quality, trust, to name just a few.

#### D. Research Statement

There are a few challenging questions arising from the above-mentioned scenario. Obviously, the elasticity of the NFPs of the consulting services is largely driven by human resources (e.g. number of networked consultants, their expertise and their availability) and IT resources (e.g. the quality of in-house multimedia and in-house chat, the readiness of off-the-shelf components and software applications sourced from clouds) being utilized by the consulting center. Here, the main open question is: how can we reason about the elasticity of the NFPs of a business service in tandem with IT processes, which are driven by diverse types of resources?

Furthermore, as represented in Figure 1, high-level business objectives such as  $O_1$  and  $O_2$  will influence ways in which the company arrange human resources and tune up their IT resources for their services provisioning. Constraints that matter at the business layer (e.g. coarse-grained sequence among business steps) also serve as inputs for how to efficiently provision IT resources. As such, they together put constraints on the provisioning of the consulting services. The research question here is how we model the business objectives and constraints and relate them to the elasticity of the NFPs of the business service.

This work has the following originalities.

- Correlation between the NFPs of business services and human resources may not be formulated in the same way as contemporary service discovery techniques, which attempt to match a set of required NFPs against existing NFPs at a given time.
- Correlation between the NFPs of business services and computing resources is not similar to that in contemporary business-IT alignment techniques, which try to map a business service onto a single IT process.
- It deals with business steps and business objectives at runtime and produces different IT processes for different business steps at different times.

### III. METHODOLOGY

Figure 2 illustrates the correspondence between provisioning business services and maintaining elastic processes. On one hand, the provisioning of business services is made on top of elastic processes, which are largely determined by customer demand, SCUs and the underlying computing

resources. On the other hand, the provisioning is constrained by business objectives, business constraints (e.g. coarse-grained steps between business services) and is obviously dependent on human resources.

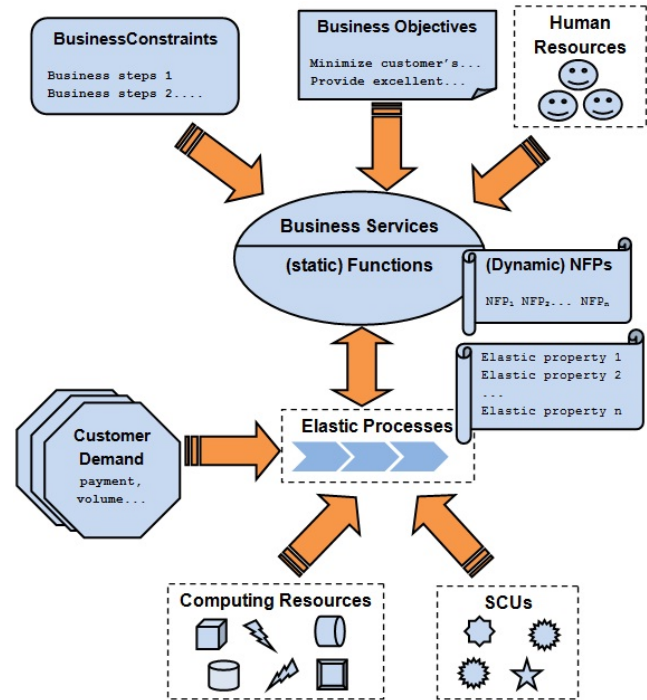


Figure 2. Business services provisioning is both elastic and constrained.

An elastic process should be equipped with an elastic reasoning mechanism that would help us determine how to utilize resources optimally [2]. To deal with business services, we need a broader framework that deals with not only elastic properties for elastic processes but also business objectives and constraints, let alone human resources. We propose the following methodology for reasoning about the elasticity and constrainedness of services provisioning.

- 1) Translate each business objective that are formulated as a goal into a logic predicate.
- 2) For each business objective that is formulated as an optimization objective, introduce a user-specified parameter in order to turn it into a predicate. For instance, minimizing customer's waiting time could practically be regarded as making sure customer's waiting time is less than four minutes.
- 3) Represent each business constraint as a fact.
- 4) Represent the elastic reasoning mechanism as a set of rules.
- 5) Specify invariants that matter on the utilization of resources.

We provide an intuitive interpretation of the elasticity and constrainedness for the example presented in Subsection II-C. Table I exemplifies how the customer demand makes

Table I  
CORRELATIONS BETWEEN BUSINESS OBJECTIVES, USER DEMAND, NFPs AND RESOURCES FOR THE EXAMPLE OF A CONSULTING CENTER

Correlation	Factor	NFP	Human resource	Computing resource
Constrained	O1: To minimize customer's waiting time	Customer's waiting time	A minimum acceptable number of experts for each consulting domain are maintained according to the forecast.	Timetable management SCU forecasts the number of customers for each consulting domain based on customer's requests in the past.
Constrained	O2: To provide excellent consulting in finance and banking domains	5-scale measurement for the quality of consulting	A large number of experts in finance and banking domains at level 3 are maintained.	Context-aware SCU gives customer-specific advices of high quality in finance and banking domains.
Elastic	Customer's request	Payment	An expert whose level and availability match customer's payment and customer's request time, respectively, is appointed.	Multimedia is loaded according to customer's preferences. How the chatting utility is sourced is determined by the customer's payment.
Elastic	Customers' preferences	Customer's satisfaction	N/A	Customers' preferences are surveyed (and retrieved from customers' info). Multimedia stores are tuned up accordingly to best serve waiting customers.
Elastic	Consulting domains	The rate at which an arbitrary consulting request is successfully served.	The consulting center maintains a core team of experts whose domains are most frequently requested.	AI chat utility is geared up to the frequently requested domains.
Elastic	Customers looking for non-human consulting	Payment & Customer's satisfaction	The team of experts maintained by the consulting center grow (or shrink) if fewer (or more) customers look for non-human consulting.	Multimedia stores and chatting facilities should be played down (or scaled up) if fewer (or more) customers look for non-human consulting.
Constrained	C2P consulting always takes place before P2P consulting does	N/A	Human resources (e.g. coordinators) engaging in Timetable management SCU are present 24/7. Experts will come later.	Multimedia stores and chatting facilities are available 24/7.

the consulting services provisioning elastic and how business objectives & constraints may influence NFPs of the consulting services and ways in which resources are deployed in the consulting center. Each row represents a correlation, which is of either elasticity or constrainedness – indicated by the left most column. Other columns from left to right are: objective/constraint or customer demand, NFPs, human resources utilized, computing resources deployed.

#### IV. MODELING AND REASONING TECHNIQUES

Based on our presented methodology, in this section we devise a framework for reasoning about the elasticity and the constrainedness.

##### A. Formal Representations

1) *Non-Functional Properties*: We represent NFPs and their quantifiable ranges using semiring. A semiring is a mathematical structure that features a domain plus two operations satisfying certain properties, as defined in Definition 1. An idempotent semiring is a semiring whose additive operation is idempotent (i.e.  $a \oplus a = a$ ). This idempotence property allows us to endow a semiring with a canonical order defined as  $a \leq b$  iff  $a \oplus b = b$  [9]. There exists another form of idempotent semiring called c-semiring whereby the  $\oplus$  operator is defined over subsets of a domain and as such it has flattening property [10]. The endowed order of a c-semiring is actually a partial order that would be used

for choosing “best” solutions in a constraint satisfaction problem.

**Definition 1.** A semiring is a tuple  $\langle A, \oplus, \otimes, 0, 1 \rangle$  such that

- $A$  is a set and  $0, 1 \in A$
- $\oplus$ , called the additive operation, is a commutative, associative operation having 0 as its unit element (i.e.  $a \oplus 0 = a = 0 \oplus a$ )
- $\otimes$ , called the multiplicative operation, is an associative operation such that 1 is its unit element and 0 is its absorbing element (i.e.  $a \otimes 0 = 0 = 0 \otimes a$ )
- $\otimes$  distributes over  $\oplus$  (i.e.  $\forall a, b, c \in A \rightarrow a \otimes (b \oplus c) = a \otimes b \oplus a \otimes c$ )

**Example 1.** The waiting time customers have to spend before actually talking to an expert can be expressed as semiring  $\langle \mathbb{R}^+, \min, +, \infty, 0 \rangle$  with  $\min(a, b)$  and classical addition serving as the (idempotent) semiring additive operation and the semiring multiplication operation, respectively.

**Example 2.** The quality of consulting a customer receives may be expressed as semiring  $\langle \{EL, VG, GD, FR, PR\}, \max, \min, PR, EL \rangle$  with  $\max(a, b)$  and  $\min(a, b)$  serving as the (idempotent) additive operation and the multiplication operation of the semiring, respectively.  $EL, VG, GD, FR$  and  $PR$  stand for the 5-scale rating (i.e. excellent, very good, good, fair and poor).

A bounded lattice has a greatest element (denoted as  $\top$ ) and a least element (denoted as  $\perp$ ) and features two operations: meet (denoted as  $\wedge$ ) and join (denoted as  $\vee$ ) [9]. To represent a NFP and its acceptable elasticity, we first pick an appropriate semiring and then specify a lattice on the selected semiring.

2) *Business Objectives*: Business objectives as presented in Subsection II-B can be formalized as follows, assuming  $L_{nfp} = \{p_1, p_2 \dots p_n\}$  denotes a predicate that refers to a list of NFPs of the business services being provided.

- A goal is a tuple  $\langle G_{kind}, L_{nfp} \rangle$  where  $G_{kind}$  is either *achieve*, *avoid*, *maintain* or *cease* according to a widely-accepted goal taxonomy [11].
- An objective function is a tuple  $\langle F, L_{nfp} \rangle$  where  $F$  is either *maximization* or *minimization*;
- A plan is a sequence of goals  $\{G_1, G_2 \dots G_n\}$ .

3) *Resources*: We need to represent human resources and their availability. The availability of a person (as a resource) can be expressed as: on/off business hours or unavailable (e.g. on vacation). Formally, a human resource is a tuple  $\langle cap, level, f_{avail} \rangle$  such that

- $cap \in C$  where  $C$  is the set of all capabilities
- $level \in \mathbb{N}$  is the expertise level of the person representing the human resource in question. Three or four levels usually suffice in most cases.
- $f_{avail} : Date \rightarrow \{BH, OH, NA\}$  is a function that tells how the person in question is available on a given day.  $BH$ ,  $OH$  and  $NA$  correspond to on-business hours, off-business hours and unavailability.

The cost of hiring a human resource correlates with her/his expertise level and the period of time during which she/he is hired (i.e. the higher level an expert is at, the better payment she/he receives from the consulting center; hiring an expert at off-business hours is generally more expensive).

As for the availability of an SCU we come up with another tuple  $\langle func, q, f_{card} \rangle$  such that

- $func$  denotes the functionality of the SCU being represented
- $q$  denotes the quality
- $f_{card} : Time \rightarrow \mathbb{N}$  is a function that yields the maximum number the SCU being represented can be invoked at a given time.

### B. Reasoning and Mapping

We can formally reason about the elasticity and constrainedness of business services by turning them into a constraint satisfaction problem. For this purpose, we use a mathematical structure called c-semiring<sup>4</sup> to handle the constraints.

Figure 3 gives an overview of our to-be reasoning system that helps relate business services with IT elastic processes.

<sup>4</sup>A c-semiring (c- stands for constraints) is similar to a semiring but its  $\oplus$  matters over a subset of elements and has flattening property [10].

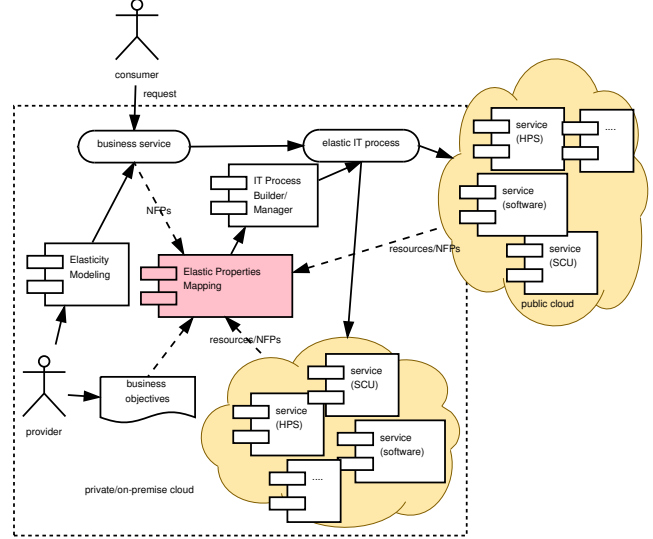


Figure 3. System component view

Essentially, it will be implemented by components *Elasticity Modeling* and *Elastic Properties Mapping*. Technically, our system works with clouds of services by considering private and public clouds for IT services. The provider's resources can be put in the private cloud. For example, the provider may have some consultants in house. Combining both private and public clouds is essential for mapping as a provider always needs to utilize its private resources first, and then obtains public pay-per-use resources, if on-premise resources are not enough.

### V. PROTOTYPE AND SIMULATION

To validate our concepts, we take a simulation approach. All of the predicates, rules, facts and invariants that factored out in these steps are considered as constraints. We feed them to a solver that can find solutions. To this end, we develop simulations of our framework using Alloy - a formal declarative language based on first-order logic and set theory [5]. This language is capable of processing first-order statements and comes with an Object-Oriented syntax. This allows us to declaratively represent all concepts that are formalized in Subsection IV-B and captures all constraints in a single module of Alloy code.

Figure 4 is a fragment of Alloy code that represents human resources, computing resources, customer's consulting requests (that have NFPs) and other minor concepts (e.g. payment, date, time). They are declared using Alloy keyword `sig`. This keyword signifies Alloy *signature*, which is the counterpart of the *class* construct in object-oriented programming. Each signature has a few Alloy fields each if which is declared as a set and is accompanied by an Alloy keyword specifying its cardinality (e.g. `lone` and `one`). In particular, field `availability` of `HumanResource` is

```

sig CalendarDate {
  gregorian: one String
}

sig Time {
  date: one CalendarDate,
  hour: one Int
} { hour >= 0 and hour < 23 }

abstract sig AvailabilityOnDate {}

one sig OnBizHours, OffBizHours extends AvailabilityOnDate {}

abstract sig Expertise {}

one sig Finance, Cars, Family extends Expertise {}

sig HumanResource {
  level: one Int,
  domain: one Expertise,
  availability: AvailabilityOnDate -> CalendarDate
} {
  level = 1 or level = 2 or level = 3
}

abstract sig ComputingResource {
  quality: one String
} {
  quality = "High Quality" or quality = "Low Quality"
}

one sig AIChat, TimetableMgt, ContextAwareAdvice
  extends ComputingResource {}

abstract sig Payment {}

one sig Premium, Intermediate, EntryLevel
  extends Payment {}

sig ConsultingRequest {
  consultee: one String,
  rtime: one Time,
  payment: one Payment,
  operator: lone String,
  consultant: lone HumanResource,
  its: seq ComputingResource,
} {
  #its > 0
  (some consultant and rtime.hour > 8 and rtime.hour < 17)
  => OnBizHours in consultant.availability.(rtime.date)
  (some consultant and (rtime.hour <= 8 or rtime.hour >= 17))
  => OffBizHours in consultant.availability.(rtime.date)
}

```

Figure 4. Alloy signatures that model resources utilized for provisioning consulting services.

declared as a set of relations, which semantically expresses a query on how available a person is on a calendar date. In some cases, we see a block structure right after the declaration of an Alloy signature. This block lists Alloy facts - invariants that matter on the very signature being declared.

Figure 5 is another fragment of Alloy code that captures business objectives O1 and O2, the constraint stating that C2P always takes place before P2P and an elasticity mechanism to determine the expertise level (of the expert appointed) assigned to and the quality of computing services allocated for a customer's request. The two objectives are encoded using the `pred` construct of Alloy. The constraint is expressed under an Alloy fact named C2PB4P2P. In this Alloy fact, symbol # refers to the cardinality of the field (as a

```

// determine the level of expertise and quality of software services
fact elasticity {
  all cr: ConsultingRequest | (cr.payment = Premium and some cr.consultant)
  => cr.consultant.level >= 3
  all cr: ConsultingRequest | (cr.payment = Intermediate and some cr.consultant)
  => cr.consultant.level >= 2
  all cr: ConsultingRequest | (cr.payment = EntryLevel and some cr.consultant)
  => cr.consultant.level >= 1
  all cr: ConsultingRequest | all c: cr.its.elems |
  cr.payment = Premium => c.quality = "High Quality" else c.quality = "Low Quality"
}

// C2P consulting takes place before P2P consulting
fact C2PB4P2P {
  all r: ConsultingRequest | #r.consultant > 0 => #r.its > 1
}

// To provide excellent consulting in finance and banking domains
pred O2 {
  all e: HumanResource | e.domain in Finance => e.level = 3 else e.level < 3
}

// To minimize customer's waiting time
pred O1 {
  all cr: ConsultingRequest | #cr.its < 4
}

```

Figure 5. Alloy facts and predicates for reasoning about the elasticity and constrainedness of provisioning consulting services.

set) being referenced. The elasticity mechanism is expressed under another fact that has self-explanatory Alloy facts.

To verify and execute our models in Figures 4 and 5, we use the Alloy Analyzer tool<sup>5</sup>. The tool first serves as a compiler to check if the Alloy code is syntactically correct and then acts as an interpreter to verify if the code is semantically consistent (i.e. not overconstrained) and simulate it. Figure 6 gives a visualization of an execution of this Alloy model. In overall, this prototype and its simulation confirm the applicability of our methodology and framework.

## VI. RELATED WORK

Web service selection and adaptation techniques usually consider to replace problematic services with similar services (with better NFPs) [12], [13] but they do not consider the mapping from business steps to elastic IT processes. Service techniques have been developed for selecting and optimizing workflows based on NFPs of software services [14]. Our work considers both software and human-based services not for the business service as a whole but only for individual business steps at runtime.

Runtime resource mapping and provisioning in the cloud has also attracted several work. There are two major differences between our work and related work. First, they either consider only software-based resources [15], [16] or human-based resources, while we consider both in the same business service. Secondly, related work is focus on limited NFPs [16] but do not consider business strategies at runtime.

Existing cloud platforms have provided elastic commands for scaling in/out resources but contemporary elastic processes take only machine and software resources [15]. Conceptually, resource mapping algorithms treat resources uniformly regardless of what pools they are from. But

<sup>5</sup>Alloy homepage <http://alloy.mit.edu/alloy/>

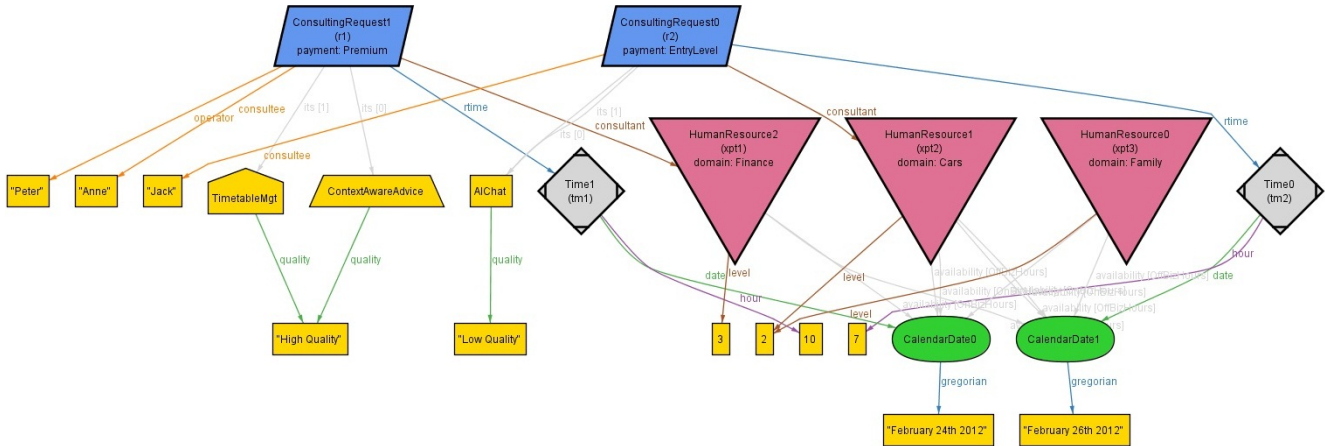


Figure 6. Visualizing an execution of the Alloy model presented in Figures 4 and 5.

in practice, there are substantial differences between local resources and remote resources. Regarding this point, our framework ensures that resources for our business steps to be elastic from on-premise to third-party resources.

## VII. CONCLUSION

In the context of SoEA, business services provisioning is driven by many factors including resources, business objectives, business constraints and the customer demand. Essentially, the provisioning is made on top of elastic IT processes, which are strained by both the resources and the customer demand. The provisioning is also constrained by the business objectives and, to some extent, coarse-grained business constraints. In this paper, we present a novel framework for reasoning about this elasticity and constraintness. We propose a methodology, devise an engine to find possible service provisioning solutions and provide a simulation using Alloy.

Work is currently underway to develop techniques for harvesting best solutions. Another direction of future work would be to address additional NFPs including costs and service contracts in our reasoning framework.

## REFERENCES

- [1] S. Dustdar and H.-L. Truong, "Virtualizing software and humans for elastic processes in multiple clouds – a service management perspective," *International Journal of Next-Generation Computing (IJNGC)*, 2012, to appear.
- [2] S. Dustdar, Y. Guo, B. Satzger, and H.-L. Truong, "Principles of Elastic Processes," *IEEE Internet Computing*, vol. 15, no. 5, pp. 66–71, 2011.
- [3] L.-S. Lê, A. Ghose, M. Krishnan, K.-M. Krishnankunju, and K. Hoesch-Klohe, "Correlating Business Objectives with Services: An Ontology-Driven Approach," in *Proceedings of the 8<sup>th</sup> IEEE International Conference on Services Computing*. IEEE Computer Society, July 2011, pp. 306–313.
- [4] L.-S. Lê, H. Dam, and A. Ghose, "On Business Services Representation – The 3 x 3 x 3 Approach," in *Proceedings of 21<sup>st</sup> Australasian Conference on Information Systems*. Brisbane, Australia: Association for Information Systems, December 2010.
- [5] D. Jackson, *Software Abstractions: Logic, Language, and Analysis*. Cambridge, MA: MIT Press, 2006.
- [6] P.-P. Maglio, S. Srinivasan, J.-T. Kreulen, and J. Spohrer, "Service Systems, Service Scientists, SSME, and Innovation," *Communications of the ACM*, vol. 49, no. 7, pp. 81–85, July 2006.
- [7] M. Lankhorst, *Enterprise Architecture at Work - Modelling, Communication and Analysis*, 2nd ed. Springer Berlin Heidelberg, 2009.
- [8] W. Engelsman, D. Quartel, H. Jonkers, and M. van Sinderen, "Extending Enterprise Architecture Modelling with Business Goals and Requirements," *Enterprise Information Systems*, vol. 5, no. 1, pp. 9–36, February 2011.
- [9] B.-A. Davey and H.-A. Priestley, *Introduction to Lattices and Order*. Cambridge University Press, 2002.
- [10] S. Bistarelli, U. Montanari, and F. Rossi, "Semiring-based constraint satisfaction and optimization," *Journal of the ACM*, vol. 44, no. 2, pp. 201–236, March 1997.
- [11] A. Dardenne, A. van Lamsweerde, and S. Fickas, "Goal-directed requirements acquisition," *Science of Computer Programming*, vol. 20, no. 1-2, pp. 3–50, April 1993.
- [12] T. Yu and K.-J. Lin, "A Broker-based Framework for QoS-aware Web Service Composition," in *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service, 2005*. IEEE Computer Society, March 2005, pp. 22–29.
- [13] V. Cardellini, E. Casalicchio, V. Grassi, F. Lo Presti, and R. Mirandola, "QoS-Driven Runtime Adaptation of Service-Oriented Architectures," in *Proceedings of the the 7<sup>th</sup> joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering*. ACM, 2009, pp. 131–140.
- [14] G. Canfora, M. D. Penta, R. Esposito, and M. L. Villani, "An Approach for QoS-aware Service Composition Based on Genetic Algorithms," in *Proceedings of the 2005 conference on Genetic and evolutionary computation*, 2005, pp. 1069–1075.
- [15] T. Dornemann, E. Juhnke, and B. Freisleben, "On-Demand Resource Provisioning for BPEL Workflows Using Amazon's Elastic Compute Cloud," in *Proceedings of the 9<sup>th</sup> IEEE/ACM International Symposium on Cluster Computing and the Grid*, May 2009, pp. 140–147.
- [16] C. Vecchiola, R.-N. Calheiros, D. Karunamoorthy, and R. Buyya, "Deadline-Driven Provisioning of Resources for Scientific Applications in Hybrid Clouds with Aneka," *Future Generation Computer Systems*, vol. 28, no. 1, pp. 58–65, 2012.