An Approach to Temporal-Aware Procurement of Web Services *

Octavio Martín-Díaz, Antonio Ruiz-Cortés,
Amador Durán, and Carlos Müller

Dpto. Lenguajes y Sistemas Informáticos,
ETS. Ingeniería Informática - Universidad de Sevilla,
41012 Sevilla (Spain - España)
{octavio, aruiz}@tdg.lsi.us.es
{amador, cmuller}@lsi.us.es

Abstract. In the context of web service procurement (WSP), temporal–awareness refers to managing service demands and offers which are subject to validity periods, i.e. their evaluation depends not only on quality of service (QoS) values but also on time. For example, the QoS of some web services can be considered critical in working hours (9:00 to 17:00 from Monday to Friday) and irrelevant at any other moment. Until now, the expressiveness of such temporal–aware specifications has been quite limited. As far as we know, most proposals have considered validity periods to be composed of a single temporal interval. Other proposals, which could allow more expressive time–dependent specifications, have not performed a detailed study about all the underlying complexities of such approach, in spite of the fact that dealing with complex expressions on temporality is not a trivial task at all. As a matter of fact, it requires a special design of the so–called *procurement tasks* (consistency and conformance checking, and optimal selection). In this paper, we present a constraint–based approach to temporal–aware WSP. Using constraints allows a great deal of expressiveness, so that not only demands and offers can be assigned validity periods but also their conditions can be assigned (possibly multiple) validity temporal subintervals. Apart from revising the semantics of procurement tasks, which we previously presented in the first edition of the ICSOC conferences, we also introduce the notion of the *covering set of a demand*, a topic which is closely related to temporality.

Keywords: services, procurement, quality, temporality, constraint programming.

1 Introduction

Web service procurement (WSP)—including automated search and selection—of the best web services according to their offered quality of service (QoS) is an activity which is gaining importance in the development of enterprise–level systems with a service–oriented architecture (SOA) [18, 24].

* This work has been funded by the Spanish Government under grant TIC2003-02737-C02-01, AGILWEB project.

B. Benatallah, F. Casati, and P. Traverso (Eds.): ICSOC 2005, LNCS 3826, pp. 170–184, 2005.

Web services, as a particular case of *software packages*, must be selected according to user requirements [3, 4]. On the one hand, these user requirements, to which we refer to as *demands*, are usually specified using boolean expressions, i.e. conditions on attributes describing the desired QoS of a service, for example[1] $MTTF \geq 100$. On the other hand, web service providers usually guarantee the QoS of the service they provide, i.e. their *offers*, for example $100 \leq MTTF \leq 120$.

Procurement is the process of finding the best offer for a given demand [18]. Its typical scenario is: (1) a provider advertises its offers in a repository, (2) a customer asks its matchmaker for an offer to meet its demands, and (3) the matchmaker searches for matching offers, returning a result which may be an optimal offer according to a given customer criterion, or a failure message if no matching offer is found [21].

Temporality is an important aspect of WSP. If a demand or offer is subject to a validity period, it is said to be *temporal–aware*. As an example, in order to specify a (part of a) demand as "*the MTTF of the web service at working hours (9:00 to 17:00, Monday to Friday) should be (at least) of 99%, otherwise 90%*", we would require to define multiple, periodical validity periods associated to concrete conditions of the demand. Other temporal aspects to be taken into consideration are the granularity of time points, periods and durations, and the different time zones in which demands and offers (D&O) can be available.

Not only it is necessary to extend the current models in WSP in order to improve their expressivess regarding temporality, but it is also needed to re–think the so–called procurement tasks, i.e. consistency and conformance checkings, and optimal selection, because of the non-trivial, intrinsic semantics of temporal expressions. For example, in a non-temporal-aware context we define the notion of *pessimistic conformance* so that an offer is conformant to a demand iff all the quality values guaranteed by the offer satisfy the conditions imposed by the demand. Let imagine a dummy demand and offer which were constituted by only a validity period, with no conditions regarding any quality attribute. If the validity period of the offer were included in the validity period of the demand, then such offer could be considered as conformant. But this is not the case, because the offer does not cover the validity period of the demand, so the offer is not conformant. In general, if temporality is taken into account, the notions of consistency, conformance, and optimal selection must be revised.

Until now—to the best of our knowledge— proposals allow a demand or an offer to have a validity period composed of only a single temporal interval. Only a few of them allows more complex temporal expressions, but most of them have not provided a detailed study about the underlying complexities of operations due to temporal semantics.

In this paper, we present an approach to temporal–aware WSP which is based on constraint programming (CP). It is based on notions introduced in our previous non–temporal–aware, constraint–based approach to WSP [15, 18]. Using CP for WSP entails some advantages. First, D&O can be stated declaratively, endowing the symmetric model with a very powerful expressiveness so that D&O can be specified with the same expressiveness. Thus, offers are not limited to single parameter–value pairs. Moreover,

[1] MTTF stands for *mean time to failure*.

```
// Service Demand for IVideoServer

using Reliability, Hosting;
product IVideoServer;

valid zone GMT +1 {
  global { during 01/JUN/2005..31/AGO/2005; }
   WORKING { from 9..17 on MON..FRI; }
  HOME { global except WORKING; }
  SEASON { during 15/JUL/2005..15/ AGO/2005;
}
}

requires {
  D1: MTTF ≥ 100 and MTTR ≤ 10 on WORKING;
  D2: MTTF ≥ 90 and MTTR ≤ 15 on HOME;
  D3: COST ≤ 10 on SEASON;
}

guarantees {
  D4: HOST = SPAIN;
}

assessment {
  MTTF { importance = VERY_HIGH, {
     WORKING { (0,0), (80,0), (100,0.5), (120,1)
};
    HOME { (0,0), (60,0), (90,0.5), (120,1) }; }
  }
  MTTR { importance = LOW, {
    global { (0,1), (5,1), (10,0.5), (15,0) }; }
  }
}
```

Fig. 1. An example of temporal-aware demand written in QRL

it is not necessary to write specific procedures for procurement tasks because they are implemented by checking properties of D&O by means of a constraint solver.

Figure 1 shows an illustrative example of a temporal-aware demand. It is written in QRL (*Quality Requirements Language*), which is a language specifically devised for that purpose by one of the authors of this work as part of his PhD thesis [17]. This example is intended to be self–explanatory, in order to give an overview of the expressiveness of our approach. First, the demand establishes the Central Europe time zone (UTC/GMT+1). Then, it defines the global validity period (VP) together with other validity periods. The *working hours* VP is composed of some periodical temporal intervals, whereas the *home hours* VP is computed from the global VP and the previous one. Another valididy period is *season* which is non-periodical.

Note the validity periods can be assigned to conditions of the demand, so that the conditions on the same quality attributes are different at *working hours* or *home hours*. The *season* VP indicates the dates between which the cost of using the service should not be greater than 10 €. The demand's host is always in Spain at any time of the global VP.

Note also the assessment criteria include utility functions which depends upon time. These functions are defined in a piecewise–like way. Each point is associated to the corresponding utility value (between 0 and 1), so that two consecutive points form a segment of the function. Utility functions are weighted by their grades of importance.

We also introduce the notion of *covering*. Since it is possible that none of the available offers were conformant to a given demand because they did not cover it, one could think of selecting several offers which are grouped together, in order to *build* a conformant offer which covers the validity period of the demand.

The rest of the paper is structured as follows. First, Section 2 introduces the theoretical basis for interpreting the temporal-aware procurement tasks by means of CSP, so that Section 3 presents our proposal to model them. Next, Section 4 provides a review of the state–of–the–art. Finaly, Section 5 concludes the paper and presents the future work.

2 Constraint Programming in a Nutshell

Constraint programming (CP) is the study of computational models and systems based on contraints. CP is becoming a very interesting alternative to the modeling of optimization problems because of its potential to solve hard, real–life problems, and its declarative nature. A problem expressed as a set of constraints is formalized as a *contraint satisfaction (optimization) problem* (CSP) [5, 7, 8].

2.1 Basic Definitions

In this section, we introduce CP as the underlying formalism of our approach for expressing D&O. The core of our proposal was a set of definitions used to rigorously define the so–called procurement tasks.

Definition 1 (CSP). *A CSP is a three–tuple of the form (V, D, C) where $V \neq \emptyset$ is a finite set of variables, $D \neq \emptyset$ is a finite set of domains (one for each variable) and C is a set of constraints defined on V.*

For instance, for the following CSP $(\{x, y\}, \{[0..2], [0..2]\}, \{x+y < 4, x-y \geq 1\})$, the assignment $\sigma = \{x \mapsto 2, y \mapsto 0\}$ is one of its solutions.

Definition 2 (Solution Space). *Let ψ be a CSP of the form (V, D, C), its solution space, denoted as* $\mathrm{sol}(\psi)$*, is composed of all its possible solutions.*

$$\mathrm{sol}(\psi) = \{\ \sigma \in V \to D \mid \sigma(C)\ \}$$

where $\sigma(C)$ holds iff each assignment in σ satisfies every constraint in C.

In the previous example the solution space is $\{\{x \mapsto 1, y \mapsto 0\}, \{x \mapsto 2, y \mapsto 0\}, \{x \mapsto 2, y \mapsto 1\}\}$.

Definition 3 (Satisfiability). *Let ψ be a CSP of the form (V, D, C), ψ is said to be satisfiable, denoted as* $\mathrm{sat}(\psi)$*, iff its solution space is not empty.*

$$\mathrm{sat}(\psi) \Leftrightarrow \mathrm{sol}(\psi) \neq \emptyset$$

Definition 4 (Minimum Space and Value). *Let ψ be a CSP of the form (V, D, C), its minimum space with regard to an objective function O, denoted as* $\min_S(\psi, O)$*, is composed of all the solutions of ψ that minimize O. Its minimum value with regard to O, denoted as* $\min_V(\psi, O)$*, is the value the objective function takes on* $\min_S(\psi, O)$*.*

$$\min_S(\psi, O) = \{\ \sigma \in \mathrm{sol}(\psi)\ \mid\ \forall\, \sigma' \in \mathrm{sol}(\psi)\ \cdot\ O(\sigma) \leq O(\sigma')\ \}$$
$$\min_V(\psi, O) = m\ \Leftrightarrow\ \forall\, \sigma \in \min_S(\psi, O)\ \cdot\ O(\sigma) = m$$

For instance, consider the CSP in the previous example and an objective function defined as $O(x, y) = x^2 y$. In this case, $\min_S(\psi, O) = \{\{x \mapsto 1, y \mapsto 0\}, \{x \mapsto 2, y \mapsto 0\}\}$. The minimum value is 0.

2.2 Filters and Projections

In general, the solution space of a CSP can be restricted by means of intersecting a second CSP.

Filters. A filter is a kind of selection, which allows to obtain a CSP whose solution space has been restricted to those solutions containing a (possibly partial) assignment over the variables.

Definition 5 (Filtering). *Let ψ be a CSP of the form (V, D, C), and $\sigma_\pi = \{v_1 \mapsto d_1, \ldots, v_k \mapsto d_k\}$ an assignment defined over the k variables in $\pi \subseteq V$, the filtering of ψ on σ_π, denoted as $\psi_{v_1 \mapsto d_1, \ldots, v_k \mapsto d_k}$, is another CSP defined on V and D whose constraint set C' is C wherein as many equality constraints as assignments in σ_π have been added.*

$$C' = C \cup \bigcup_{i=1}^{k} \{v_i = d_i\}$$

In the previous example, the filtering over $\sigma_\pi = \{y \mapsto 0\}$ results in a CSP whose solution space is $\{\{x \mapsto 1, y \mapsto 0\}, \{x \mapsto 2, y \mapsto 0\}\}$.

Projections. A projection is another kind of selection, which allows to obtain those values which take a set of variables whenever the CSP is satisfiable.

Definition 6 (Projection). *Let ψ be a CSP of the form (V, D, C), and π a set of variables such that $\pi \subseteq V$, the projection of ψ over π, denoted as $\psi_{\Downarrow\pi}$, is another CSP defined on π and D_π whose solution space is composed of values of variables in π which are part of any solution in* $\mathrm{sol}(\psi)$.

$$\mathrm{sol}(\psi_{\Downarrow\pi}) = \{ \sigma_\pi \in \pi \to D_\pi \mid \exists\, \sigma \in \mathrm{sol}(\psi) \cdot \sigma_\pi \subseteq \sigma \}$$

where $D_\pi \subseteq D$ is the set of domains of variables in π.

In the previous example, the projection of the solution space over $\pi = \{x\}$ results in $\{\{x \mapsto 1\}, \{x \mapsto 2\}\}$.

3 Temporal-Aware Procurement Using Constraint Programming

In [15, 18], we described how CP can help automating the procurement tasks, i.e. the checking for consistency and conformance, and selection of optimal offers. The key to automating the procurement tasks is to map D&O onto CSPs. In order to do so, each attribute must be mapped onto a variable with its corresponding domain, and each condition must be mapped onto a constraint.

In this section, we review these notions in order to make them temporal-aware. We assume a linear, discrete time-structure based on natural numbers. Time elements are point times and temporal intervals. A temporal interval is given by two time points representing their extremes.

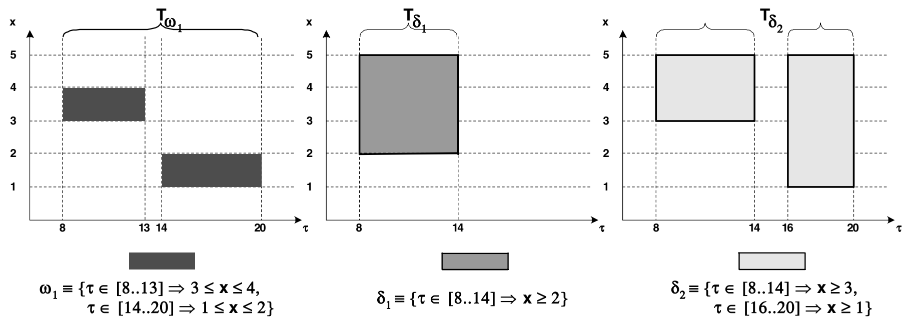

Fig. 2. Solution spaces of temporal-aware offers and demands

3.1 Demands and Offers

Demands assert the conditions the provider shall meet, whereas offers assert the conditions a provider guarantees[2]. Regarding temporality, all D&O are considered as (by default) temporal-aware, i.e. they all have a *validity period* and their inner conditions can (optionally) establish time-dependent demand requirements or offer guarantees. If a D&O does not have an explicit validity period, it will be supposed to have an infinite temporal interval.

Let δ denote a demand, and ω denote an offer. Their corresponding CSP are denoted as ψ_δ and ψ_ω, respectively. Let α denote a demand or offer. Any demand or offer α has an (implicit) temporal variable, denoted as τ, so that its domain D_τ corresponds to the validity period. Inner conditions of D&O are based on QoS attributes and (eventually) the temporal variable, so that distinct temporal subintervals can be assigned to them, provided these subintervals are included in the validity period.

T_α stands for a CSP of the form $(\tau, D_\tau, true)$ whose its solution space corresponds to the validity period of α. Note $\tau' \in T_\alpha$ is a shorthand for an assignment at time τ' which belongs to the validity period.

For instance, the following tuples denote an offer ω_1 and two demands δ_1 and δ_2:

$$\omega_1 = (\{x,\tau\}, \{[0..5], [8..20]\}, \{\tau \in [8..13] \Rightarrow 3 \le x \le 4, \tau \in [14..20] \Rightarrow 1 \le x \le 2\})$$
$$\delta_1 = (\{x,\tau\}, \{[0..5], [8..14]\}, \{\tau \in [8..14] \Rightarrow x \ge 2\})$$
$$\delta_2 = (\{x,\tau\}, \{[0..5], [8..14] \cup [16..20]\}, \{\tau \in [8..14] \Rightarrow x \ge 3, \tau \in [16..20] \Rightarrow x \ge 1\})$$

Their solution spaces are shown graphically in Figure 2. The offer ω_1 has the temporal interval $[8..20]$ as validity period, representing the office hours of a day. Each guarantee of this offer is assigned a temporal subinterval which is included in the validity period, covering the overrall temporal interval. The first guarantee of ω_1 is valid at times in $[8..13]$. The second guarantee of ω_1 is valid at times in $[14..20]$.

Note the "$\Rightarrow$" operator is the logic implication with its usual meaning.

[2] For the sake of simplicity, we are assuming a one-way matchmaking, i.e. demands only require something from offers, and offers only guarantee something to demands, but not viceversa. The interested reader is referred to [18] wherein a two-way matchmaking is presented.

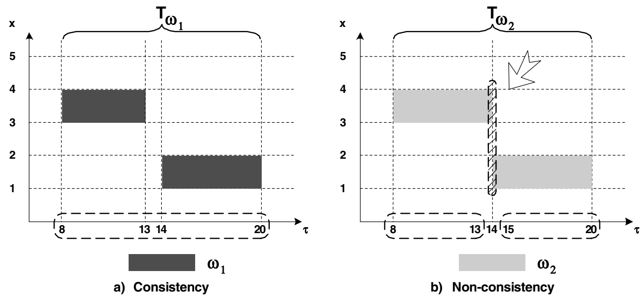

Fig. 3. Temporal-aware consistency

The first demand δ_1 has a unique requirement whose temporal subinterval is regarded to the overall validity period $[8..14]$. This demand is not defined at any other time of a day.

The second demand δ_2 has a validity period composed of two subintervals $[8..14]$ and $[16..20]$ so that each requirement is assigned to every subinterval.

3.2 Consistency

Checking a demand or offer for consistency allows to unveil whether they have internal contradictions or not along the times whenever it is defined. If temporality is taken into account, consistency must also involve a checking of their validity periods. Moreover, since their requirements or guarantees can be also assigned one or more temporal intervals, they should be included in the validity period in order to be considered as consistent.

Note that it is also possible different demand requirements or offer guarantees have to be fulfilled at the same time. Checking the consistency of conditions and validity periods separately is not enough, but once validity periods have been checked, the consistency of conjunction of all demands requirements or offer guarantees at any time of the validity period has to be checked, as well.

Definition 7 (Consistency). *A demand or offer α is said to be consistent iff the projection over time of its corresponding CSP ψ_α equals its non-empty validity period.*

$$consistent(\alpha) \Leftrightarrow sol(\psi_{\alpha \Downarrow \tau}) = sol(T_\alpha)$$

For instance, consider the offer ω_1 in the previous example, and another offer ω_2 defined on the same attributes and domains but with the following conditions:

$$\{\tau \in [8..14] \Rightarrow 3 \leq x \leq 5, \tau \in [14..20] \Rightarrow 1 \leq x \leq 3\}$$

Both of them are shown in Figure 3. Note that the offer ω_1 is consistent (see Figure 3.a) because there are no contradictory conditions at any time in the validity period. However, the offer ω_2 is not consistent (see Figure 3.b) because at time $\tau = 14$ (marked with an arrow) there exist two contradictory conditions, so that the solution space of their conjuction at such a time is empty, and that point time is not included in the projection. Therefore, since the projection does not equal the validity period, the offer ω_2 is not consistent.

3.3 Conformance

Checking if an offer conforms to a demand allows to know whether the values guaranteed by a party (the offer from a provider) meet the values required by the other party (the demand of a client) whenever the demand is defined. A non-temporal-aware offer ω and a non-temporal-aware demand δ is said to be pessimistic-conformant iff the solution space of ψ_ω is a subset of the solution space of ψ_δ. In terms of CP, this can be expressed by means of Marriott and Stuckey expression [14]:

$$conformant(\omega, \delta) \Leftrightarrow \neg sat(\psi_\omega \wedge \neg\psi_\delta)$$

If temporality is taken into account, this checking must be carried out at any time of the validity period of the demand. In Section 1, we have introduced the need of revising the conformance notion, so that if an offer and a demand were defined exclusively by their validity periods, then they would be considered as conformant iff the validity period of the offer covered the validity period of the demand.

Definition 8 (Conformance). *An offer ω and a demand δ are said to be conformant iff the validity period of ω covers the validity period of δ, and the projection over time of the CSP representing those solutions of ω which are not a solution of δ is disjoint to the validity period of δ.*

$$\begin{aligned} conformant(\omega, \delta) \Leftrightarrow\ & sol(T_\delta) \subseteq sol(T_\omega) \\ & \wedge sol(\{\psi_\omega \wedge \neg\psi_\delta\}_{\Downarrow\tau}) \cap sol(T_\delta) = \emptyset \end{aligned}$$

For instance, consider the offer ω_1 and the demand δ_1 in the previous example, together with the demands δ_3 and δ_4 whose definitions are:

$$\delta_3 = (\{x, \tau\}, \{[0..5], [8..20]\}, \{\tau \in [8..13] \Rightarrow x \geq 3, \tau \in [14..20] \Rightarrow x \geq 1\})$$
$$\delta_4 = (\{x, \tau\}, \{[0..5], [7..20]\}, \{\tau \in [7..13] \Rightarrow x \geq 3, \tau \in [14..20] \Rightarrow x \geq 1\})$$

Their conformance relationships are shown in Figure 4. Note that the offer ω_1 is not conformant to the demand δ_1 (see Figure 4.a) because at $\tau = 14$ (marked with an arrow) the solution space of the offer is not a subset of solution space of the demand. Note this situation is detected by the above formula, because the time $\tau = 14$ belongs to the projection over time of those solutions of ω_1 which are not included in the solution space of the demand δ_1, and it is also included in its validity period T_{δ_1}. The offer ω_1 is conformant to the demand δ_3 (see Figure 4.b) because it is conformant at any time of its validity period, covering it completely as well. Finally, the offer ω_1 is not conformant to the demand δ_4 (see Figure 4.c) because it does not cover its validity period since it does not supply anything at $\tau = 7$ (marked with an arrow). The striped zones in Figure 4 represent the solution spaces of the negated CSP corresponding to the demands.

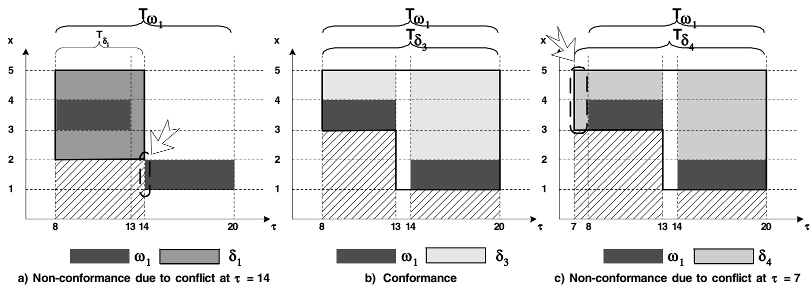

Fig. 4. Temporal-aware conformance

3.4 Finding the Optimal Offers

The final goal of matchmaking is, given a demand, finding a conformant offer that is optimal from the customer's point of view. This task is interpreted as a contraint satisfaction *optimization* problem (CSOP), which requires a preference order defined on the offer set. It is usual to establish such an order by means of a weighted composition of utility functions, whose general form is as follows:

$$\mathcal{U}(a_1, \ldots, a_n) = \sum_{i=1}^{n} k_i U_i(a_i) \quad k_i \in [0,1] \quad \sum_{i=1}^{n} k_i = 1$$

where each a_i denotes a quality attribute, each k_i its associated weight, and each U_i its associated utility function ranging over $[0,1]$ and describing how important the values of attribute are for the client.

Definition 9 (Set of Optimal Offers). *Let Ω_δ be a set of conformant offers to the demand δ, and $\mathcal{U}$ the assessment criteria given by an utility function, the set of optimal offers, denoted as $\Omega^*_{\delta,\mathcal{U}}$, is constituted of those offers in Ω_δ which maximize $\mathcal{U}$.*

$$\Omega^*_{\delta,\mathcal{U}} = \{\omega \in \Omega_\delta \mid \forall \omega' \in \Omega_\delta \cdot \mathcal{U}(\omega) \geq \mathcal{U}(\omega')\}$$

where $\mathcal{U}(\omega)$ stands for the utility of the offer ω given $\mathcal{U}$.

In a non-temporal-aware context, the utility of an offer corresponds to the worst case, that is to say, the utility of those values which minimize the utility function:

$$\mathcal{U}(\omega) = \min_V(\psi_\omega, \mathcal{U})$$

If temporality is taken into account, utility functions can be dependent upon time, so that quality attributes can have different utility values at distinct temporal intervals. The utility of an offer is the average utility during the validity period of δ:

$$\mathcal{U}(\omega) = \frac{1}{|sol(T_\delta)|} \sum_{\tau' \in T_\delta} \min_V(\psi_{\omega,\tau \mapsto \tau'}, \mathcal{U})$$

where $\psi_{\omega,\tau \mapsto \tau'}$ stands for the CSP which corresponds to ω filtered at time $\tau = \tau'$.

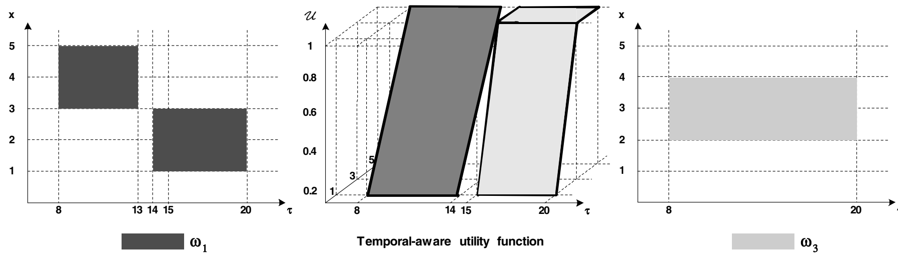

Fig. 5. Optimal selection with temporal-aware utility functions and offers

For instance, consider the offer ω_1 in the previous example, and another offer ω_3 defined on the same attributes and domains but with the following condition $\{\tau \in [8..20] \Rightarrow 2 \leq x \leq 3\}$. Assume these offers are conformant to a demand δ whose validity period is $[8..20]$, so that the assessment criteria is given by the utility function $\mathcal{U}$ in Figure 5. Note it gives different utility values for intervals $\tau \in [8..14]$ and $\tau \in [15..20]$. The set of optimal offers is $\Omega^*_{\delta,\mathcal{U}} = \{\omega_3\}$, according to their utility values:

$$\mathcal{U}(\omega_1) = \tfrac{1}{13}\{\tfrac{3}{5} \times 6 + \tfrac{1}{5} \times 1 + \tfrac{1}{3} \times 6\} = 0.45$$
$$\mathcal{U}(\omega_3) = \tfrac{1}{13}\{\tfrac{2}{5} \times 7 + \tfrac{2}{3} \times 6\} = 0.52$$

The utility of ω_1 is computed in this way. Note that the number of time points which belongs to T_δ is 13. If $\tau \in [8..13]$ (six time points) then $x = 3$ is given an utility of $3/5$, if $\tau = 14$ (one time point) then $x = 1$ is given an utility of $1/5$, and if $\tau \in [15..20]$ (another six time points) then $x = 1$ is given an utility of $1/3$. The utility of ω_3 is computed in a similar way.

3.5 Finding the Optimal Covering

Since it is possible that none of the available offers were conformant to a given demand because they did not cover it, one could think of selecting several offers so that all together are conformant to the demand, covering all the validity period. The covering problem is to find such a set of offers, optimizing according to assessment criteria from demand and other (optional) criteria, in order to adopt different strategies such as, for example, to minimize the number of offers.

Definition 10 (Covering). *Let δ be a demand and Ω a set of available offers*[3]. *Ω is said to be a covering set of δ iff there exists (at least) a conformant offer in Ω at any time of the validity period of the demand.*

$$isCoveringSet(\Omega, \delta) \Leftrightarrow \forall \tau' \in T_\delta, \exists \omega \in \Omega \cdot conformant(\omega_{\tau \mapsto \tau'}, \delta_{\tau \mapsto \tau'})$$

where $\omega_{\tau \mapsto \tau'}$ and $\delta_{\tau \mapsto \tau'}$ stand for the offer ω and the demand δ at time $\tau = \tau'$, respectively.

[3] An offer is available iff it provides the functionality required by a demand.

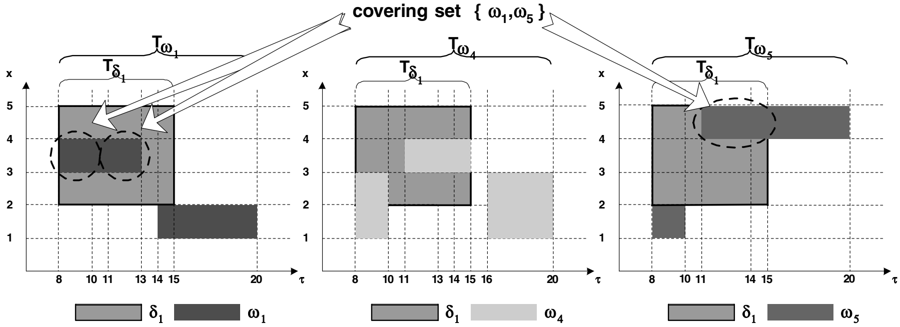

Fig. 6. Covering and temporal-awareness

For instance, consider the demand δ_1 whose valid period is $T_{\delta_1} = [8..15]$ and the assessment criteria given by the utility function $\mathcal{U}$, and the offer ω_1 in the previous example, together with the following offers ω_4 y ω_5 which are also defined on the same attributes and domains as ω_1, but with the following constraints:

$$C_{\omega_4} = \{\tau \in [8..10] \Rightarrow 1 \le x \le 3, \tau \in [11..15] \Rightarrow 3 \le x \le 4, \tau \in [16..20] \Rightarrow 1 \le x \le 3\}$$
$$C_{\omega_5} = \{\tau \in [8..10] \Rightarrow 1 \le x \le 2, \tau \in [11..20] \Rightarrow 4 \le x \le 5\}$$

The demand and offers are shown in Figure 6. None of them is conformant to the given demand, but if it were possible to join two (or more) of them, then one would be able to build a conformant offer, i.e. the so-called covering set of a demand.

Figure 6 shows the covering set of δ_1 which is constituted by the offers $\{\omega_1, \omega_5\}$ (marked with the arrows). It is a covering set because at any time of T_{δ_1} there exist (at least) a conformant offer:

$$\tau \in [8..10] \rightarrow \omega_1; \tau \in [11..13] \rightarrow \omega_1, \omega_5; \tau \in [14..15] \rightarrow \omega_5$$

The sets of offers $\{\omega_1, \omega_4\}$ and $\{\omega_1, \omega_4, \omega_5\}$ also conform a covering set of the demand. However, the set of offers $\{\omega_4, \omega_5\}$ is not a covering set, because at time $\tau \in [8..10]$ there is no offer conformant to the demand.

Among all the covering sets which can be conformed from a set of offers, one should be able to select the best one. Therefore, we need to compute their utility according to assessment criteria attached to a demand.

Let Ω be a covering set of the demand δ_1, and $\mathcal{U}$ the utility function of δ_1, the utility of a covering set is given by the aggregation of maximum utilities at any time in the validity period of δ_1:

$$\mathcal{U}(\Omega) = \frac{1}{|sol(T_\delta)|} \sum_{\tau' \in T_\delta} \max_{\omega \in \Omega_{\delta,\tau \mapsto \tau'}} \{\min_V(\psi_{\omega,\tau \mapsto \tau'}, \mathcal{U})\}$$

where $\Omega_{\delta,\tau \mapsto \tau'}$ is the subset of offers in Ω which are conformant to δ at time $\tau = \tau'$:

$$\Omega_{\delta,\tau \mapsto \tau'} = \{w \in \Omega \mid conformant(\omega_{\tau \mapsto \tau'}, \delta_{\tau \mapsto \tau'})\}$$

For instance, consider the set of offers $\Omega = \{\omega_1, \omega_4, \omega_5\}$ available to the demand δ_1, whose assessment criteria is $\mathcal{U}$, in the previous example, the utility values of the covering sets of δ_1 are:

$$\mathcal{U}(\{\omega_1, \omega_4\}) = \tfrac{1}{7}\{\tfrac{3}{5} \times 3 + \tfrac{3}{5} \times 3 + \tfrac{3}{5} \times 1 + \tfrac{3}{3} \times 1\} = 0.74$$
$$\mathcal{U}(\{\omega_1, \omega_5\}) = \tfrac{1}{7}\{\tfrac{3}{5} \times 3 + \tfrac{4}{5} \times 3 + \tfrac{4}{5} \times 1 + 1 \times 1\} = 0.86$$
$$\mathcal{U}(\{\omega_1, \omega_4, \omega_5\}) = \tfrac{1}{7}\{\tfrac{3}{5} \times 3 + \tfrac{4}{5} \times 3 + \tfrac{4}{5} \times 1 + 1 \times 1\} = 0.86$$

The utility of the covering $\{\omega_1, \omega_5\}$ is computed in this way. Note that the number of time points which belongs to T_δ is 7. If $\tau \in [8..10]$ (three time points) then the offer ω_1 has the conformant value $x = 3$ which is given an utility of $3/5$, if $\tau \in [11..13]$ (three time points) then both offers are conformant, but the best one is ω_5 because it offers a conformant value $x = 4$ which is given an utility of $4/5$ whereas ω_1 has a conformant value $x = 3$ which is given a worse utility of $3/5$, if $\tau = 14$ (one time point) then the offer ω_5 has the conformant value $x = 4$ which is given an utility of $4/5$, and if $\tau = 15$ (another one time point) then the offer ω_5 has the conformant value $x = 4$ which is given an utility of 1. The utility of the remaining coverings is computed in a similar way.

Definition 11 (Set of Optimal Coverings). *Let δ be a demand, $\mathcal{U}$ an utility function as assessment criteria, and Ω^+_δ the set of all coverings given a set of available offers. The set of optimal coverings, denoted as $\Omega^+_{\delta,\mathcal{U}}$, is constituted of those covering sets which maximize the utility function $\mathcal{U}$.*

$$\Omega^+_{\delta,\mathcal{U}} = \{\Omega \in \Omega^+_\delta \mid \forall \Omega' \in \Omega^+_\delta \cdot \mathcal{U}(\Omega) \geq \mathcal{U}(\Omega')\}$$

Given the offers and demand in the previous example, the set of optimal coverings is $\{\{\omega_1, \omega_5\}, \{\omega_1, \omega_4, \omega_5\}\}$.

Note that these covering sets have the same utility, although the latter seems to be redundant because values from ω_5 override those from ω_4. We can establish a preference order by means of any secondary assessment criteria, for example, by minimizing the number of offers. In this case, the optimal subset regarding $min_{|\Omega|}$ of $\{\{\omega_1, \omega_5\}, \{\omega_1, \omega_4, \omega_5\}\}$ is $\{\{\omega_1, \omega_5\}\}$.

4 Related Work

Figure 7 shows a brief comparison among related proposals, showing their characteristics on temporality at a first sight. Because of the limited extension of this paper, this section is devoted solely to temporal–aware proposals. A broader outline, which also includes the non–temporal–aware proposals, is available in [15, 18].

Note that our point of view is different from the perspective of service workflows, which is interested in the problem of finding an optimal execution plan of services in the context of a workflow [24]. We are interested in the procurement of web services whose demands and offers are temporal–aware. Of course, the workflow issue is very related to our problem, and they can be studied as a whole.

	Non-Periodical VP Entire Demand/Offer	Periodical VP Inner Conditions	Non-Periodical VP Inner Conditions	Multiple Intervals	Covering	Temporal Reasoning /Solving with Decidable Satisfiability
UDDIe	V					
WS-QoS	V					
WSOL		V				
WSLA			V	V		
WSML	V	V		V		
OWL-TIME	V	V	V	V		~
QRL	V	V	V	V	V	V

Fig. 7. A comparison among temporal–aware proposals

4.1 Proposals Based on Ad-Hoc Formalisms

These proposals do not have any formalism for temporal specifications, such as the *UDDI Extension* [20] and the *WS-QoS* ontology [22]. In general, they only allow to define an unique validity period for an entire demand or offer.

Fortunately, other proposals do allow to assign a validity period to every condition of a demand or an offer, such as the *IBM WSLA Web Services Level Agreement* language [6, 11] and the *WSOL Web Service Offerings Language* [23]. The *HP WSML Web Services Level Agreement Management* language [19] allows to specify both a single validity period for the entire agreement and also a periodic temporal interval to every condition. Both WSLA and WSML languages allow validity periods to be composed of multiple sub-intervals in distinct, limited ways as well.

4.2 Proposals Based on Semantic Web

These proposals are based on formalisms of the semantic web, having a much greater deal of expressiveness. The *OWL+TIME Ontology* [9, 10] is a very expressive language which is used by semantic-web-based approaches of WSP, such as the *Web Ontology Language - Services (OWL-S)* [2, 13, 16].

However, having a greater deal of expressiveness leads to several computation problems of the *Description Logics* (DL) reasoners able to reason about such temporal specifications. As a matter of fact, in logics there exist a tradeoff between expressiveness and the computability of reasoning procedures [12], so the more expressive temporal DL languages are known to be undecidable, that is to say, there is no algorithm for computing the satisfiability of a DL specification. Most of temporal DL reasoners overcome this problem by making the language less expressive, or treating the time as a concrete domain in order to use hibrid reasoners so that temporal specifications are processed by external solvers, such as the CSP solvers. In general, both (1) the reasoning on less expressive temporal DL specifications, or (2) solving a CSP, are known to be NP-complete [1].

5 Conclusions and Future Work

In this paper, we have presented an approach to add temporal–awareness to WSP by using CP, which endows our proposal with a declarative way to specify demands and offers so that the procurement tasks can be carried out by means of constraint satisfaction problems. We have introduced the notion of covering of a demand. We have also shown the need to review the semantics of procurement tasks if temporality is taken into account, and proposed a rigorous definition for them.

Our approach allows to specify a global validity period for a demand or an offer, and other validity periods which can be periodical or not, or composed of multiple intervals. These validity periods can be assigned to different conditions of the demand or the offer. Utility functions can be temporal–aware too, so that different utility values for a quality attribute can be defined at distinct time periods. The expressiveness of our approach is similar to semantic web–based proposals, though their major drawback is the undecidable nature of more complex temporal DL languages.

For future work, we are currently finishing the development of a proof–of–concept implementation, by adapting the prototype introduced in [18] so that it becomes temporal–aware. At operational level, consistency, conformance, and optimality have not to be computed at every time point of validity periods, just as they were defined in theory. A pre–processing step is needed in order to get the concrete time intervals of interest, then such tasks can be carried out on such time intervals.

Experiments need to be carried out in order to characterize the complexity of temporal–aware procurement tasks. As a result, it is expected to know what kind of temporal expressions to avoid because of their impact on the exponential behavior of CSP solving.

References

1. A. Artale and E. Franconi. A Survey of Temporal Extensions of Description Logics. *Annals of Mathematics and Artificial Intelligence*, 30(1-4):171–210, 2000.
2. The OWL Services Coalition. OWL-S: Semantic Markup for Web Services. Technical report, DARPA, 2004. `http://www.daml.org`.
3. A. Finkelstein and G. Spanoudakis. Software Package Requirements and Procurement. In *Proc. of the 8^{th} Int'l IEEE Workshop on Software Specification and Design (IWSSD'96)*. IEEE Press, 1996.
4. X. Franch and J.P. Carvallo. Using Quality Models in Software Package Selection. *IEEE Software*, 20(1):34–41, 2003.
5. E.C. Freuder and M. Wallace. Science and substance: A challenge to software engineers. *Constraints IEEE Intelligent Systems*, 2000.
6. P. Grefen, H. Ludwig, and S. Angelov. A Three-Level Framework for Process and Data Management of Complex E-services. *International Journal of Cooperative Information Systems*, 12(1):455–485, December 2003.
7. P. Hentenryck. Constraint and Integer Programming in OPL. *Informs Journal on Computing*, 14(4):345–372, 2002.
8. P. Hentenryck and V. Saraswat. Strategic directions in constraint programming. *ACM Computing Surveys*, 28(4), December 1996.

9. J. Hobbs and J. Pustejovsky. Annotating and Reasoning about Time and Events. In *Proc. of the AAAI Spring Symposium on Logical Formalization of Commonsense Reasoning*, Stanford, CA, March 2003.
10. J. Hobbs and J. Pustejovsky. An Ontology of Time for the Semantic Web. *ACM Transactions on Asian Language Processing, Special Issue on Temporal Information Processing*, 3(1):66–85, March 2004.
11. Y. Hoffner, S. Field, P. Grefen, and H. Ludwig. Contract-driven Creation and Operation of Virtual Enterprises. *Computer Networks*, (37):111–136, 2001.
12. H.J. Levesque and R.J. Brachman. Expressiveness and Tractability in Knowledge Representation and Reasoning. *Computational Intelligence*, 3(2):78–93, May 1987.
13. L. Li and I. Horrocks. A Software Framework for Mathmaking based on Semantic Web Technology. In *Proc. of the* 12^{th} *ACM Intl. Conference on World Wide Web (WWW'03)*, pages 331–339, 2003.
14. K. Marriottt and P.J. Stuckey. *Programming with Constraints: An Introduction*. The MIT Press, 1998.
15. O. Martín-Díaz, A. Ruiz-Cortés, A. Durán, D. Benavides, and M. Toro. Automating the Procurement of Web Services. In 1^{st} *Int.l Conf. on Service-Oriented Computing*, volume 2910 of *LNCS*, pages 91–103, Trento, Italy, 2003. Springer Verlag.
16. F. Pang and J. Hobbs. Time in OWL-S. In *Proc. of the AAAI Spring Symposium on Semantic Web Services*, pages 29–36, Stanford, CA, 2004.
17. A. Ruiz-Cortés. *A Semiqualitative Approach for the Automatic Management of Quality Requirements (in Spanish)*. PhD thesis, University of Seville, 2002.
18. A. Ruiz-Cortés, O. Martín-Díaz, A. Durán, and M. Toro. Improving the Automatic Procurement of Web Services using Constraint Programming. *Int. Journal on Cooperative Information Systems*, 14(4):439–467, December 2005.
19. A. Sahai, V. Machiraju, M. Sayal, L.J. Jin, and F. Casati. Automated SLA Monitoring for Web Services. Research Report HPL-2002-191, HP Laboratories, 2002.
20. A. ShaikhAli, O. Rana, R. Al-Ali, and D. Walker. UDDIe: An Extended Registry for Web Services. In *Proc. of the IEEE Int'l Workshop on Service Oriented Computing: Models, Architectures and Applications at SAINT Conference*. IEEE Press, January 2003.
21. K. Sycara, M. Klusch, S. Widoff, and J. Lu. Dynamic Service Matchmaking among Agents in Open Information Environments. *SIGMOD Record*, 28(1):47–53, 1999.
22. M. Tian, A. Gramm, T. Naumowicz, H. Ritter, and J. Schiller. A Concept for QoS Integration in Web Services. In *Proc. of the IEEE Int'l Web Services Quality Workshop (at WISE'03)*, pages 149–155, 2003.
23. V. Tosic, K. Patel, and B. Pagurek. Reusability Constructs in the Web Service Offering Language (WSOL). Research Report SCE-03-21, The Department of System and Computer Engineering, Carleton University, Ottawa, Canada, 2003.
24. L. Zeng, B. Benatallah, A.H.H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang. QoS-Aware Middleware for Web Services Composition. *IEEE Transactions on Software Engineering*, 30(5):311–327, May 2004.