Irfan ul Haq, Altaf Huqqani, and Erich Schikuta

Department of Knowledge and Business Engineering
University of Vienna, Austria
{irfan.ul.haq,huqqana3,erich.schikuta}@univie.ac.at

Abstract. Business scenarios such as Business Value Networks and E
tended Enterprises pose new challenges for service choreographies acr
heterogeneous Virtual Organizations. In such scenarios, services compo
together hierarchically in a producer-consumer manner to form serv
supply-chains of added value. Service Level Agreements (SLAs) are d
fined at various levels in this hierarchy to ensure the expected qual
of service for different stakeholders. Automation of service composit
directly implies the aggregation of their corresponding SLAs. But so f
the aggregation of SLAs has been treated only as a single layer proc
which is insufficient to complement the hierarchical aggregation of s
vices. In this paper we elaborate on the requirement of a hierarchi
aggregation of SLAs corresponding to service choreographies in Busin
Value Networks. During the hierarchical aggregation of SLAs, cert
SLA information pertaining to different stakeholders is meant to be
stricted and can be only partially revealed to a subset of their busin
partners. We introduce the concept of SLA-Views to protect such priva
concerns. We, then formalize the notion of SLA Choreography and def
an aggregation model based on SLA-Views to enable the automation
hierarchical aggregation of Service Level Agreements. The aggregati
model has been designed to comply with the WS-Agreement standar

Keywords: Service Level Agreements, Business Value Networks, Va
Chains, SLA Management.

1 Introduction

Novel concepts such as Cloud Computing, Autonomic Computing, and
Grids pursue the same industrial goal: to enable consumers to access t
resources on demand. In the notion of commodity computing, service
basic building blocks of complex software systems. A Service Level A
(SLA) is a formally negotiated contract between service provider an
consumer that ensures the expected level of service for the service c
The service consumer can be a client or another service.

In a service-enriched environment such as the Grid or the Cloud C
infrastructures, services scattered across various Virtual Organisatio
under multiple administration domains, can compose together in form

U. Dayal et al. (Eds.): BPM 2009, LNCS 5701, pp. 176–192, 2009.

services. Service composition directly implies the need of composition of corresponding SLAs. So far, SLA composition has been considered [1] a single layer process. This single layer SLA composition model is insufficient to describe supply-chain business networks. In a supply-chain, a service provider may have sub-contractors and some of those sub-contractors may have further sub-contractors making a hierarchical structure. This suppy-chain network spanned across various Virtual Organisations may emerge as a Business Value Network. Business Value Networks [2] are ways in which organizations interact with each other forming complex chains including multiple providers/administrative domains in order to drive increased business value. NESSI (Networked European Software and Services Initiative) , which is a consortium of over 300 ICT industrial partners has highlighted the importance of Business Value Networks as a viable business model in the emerging service oriented ICT infrastructure.

In addition to the notion of Business Value Networks, NESSI has pointed out various other possibilities for similar inter-organizational business models: Hierarchical Enterprises, Extended Enterprises, Dynamic Outsourcing, and so on to name a few. The process of SLA aggregation in such enterprizes is a hierarchical process. There is no SLA aggregation model till this date, which can handle this type of hierarchical aggregation. To enable these supply-chain networks in Service Oriented Infrastructures (SOI), the case of the Service Level Agreements needs to be elaborated and its issues resolved. SLA@SOI [3]is a European project that focusses on SLA issues in SOI. On its agenda is the provision of such service aggregators, that offer composed services, manageable according to higher level customer needs. In SLA@SOI's vision, service customers are empowered to precisely specify and negotiate the actual service level according to which they use a certain service.

It is not sensible to expose the complete information of SLAs spun across the whole chain of services to all the stakeholders. Not only because of the privacy concerns of the business partners, but also for disclosing it could endanger the business processes creating added value. To achieve this balance between trust and security, we introduce the concept of SLA-views. The inspiration of this concept comes from the notion of business process-views [4][5] and Workflow Views [6]. We apply the concept of views on SLA-Choreography. Each business partner will have its own view comprising of its local SLA information. The holistic effect of these views will emerge as the overall SLA-Choreography. In this paper we present a formalized approach based on the concept of SLA-Views and adherent to WS-Agreement standard, to automate the aggregation process of hierarchical SLAs in Business Value Networks. The overall contribution of this paper consists of:

- a privacy model based on the concept of SLA-Views,
- a formal description of hierarchical SLA-Choreographies based on SLA-Views in Business Value Networks,

In section 2, we give a survey of the related work. Section 3 introc hierarchical choreography of SLAs. Section 4 formalizes the concept of S and SLA Choreography. Section 5 describes the formal model of hie aggregation of SLAs and section 6 highlights some of its business app Section 7 presents a motivational example based on this model. Finally 8 concludes the paper with an overview of our achievements and strateg future work.

2 Related Work

The related work spans across three dimensions: aggregation models formal description of SLAs and the privacy of stake-holders in busines ations.

2.1 SLA Aggregation

Service Level Agreement is a contract between a service and its client; being a person or yet another service. Service composition in workflow mands SLA composition. A little research [1] [7] has been done towards SLA aggregation of workflows. Blake and Cummings [1] have defined pects of SLAs which are Compliance, Sustainability and Resiliency. Co means suitability i.e the consumer receives what is expected. Sustainabi ability to maintain the underlying services in timely fasion. Resiliency corresponds to the maintenance of services to ensure their performanc extended period of time. The authors then subdivide these three categ six aspects of SLA but this makes their approach rather specific becau not cover the whole range of SLA aspects. They put forth a model to SLAs of services mapping to a workflow but they take into account th existing only at one level. Frankova [7] has also highlighted the importar issue but has just described a vision and not any concrete model. Ung work [8]is directly relevant to our focus of research. They focus on ag of SLAs in context with Business Process Outsourcing (BPO). They syn their work with Business Process Execution Language (BPEL) and W Their model is based on SLO aggregation of SLAs on a single level. O limitation of their approach is that they take into account services relat process in one enterprise because they focus on BPO. Our approach corss-VO SLA aggregation and strictly adheres to WS-Agreement.

2.2 Formal Description of SLA

Aiello et al. [9] present a very nice formal description of SLA. Their is based on WS-Agreement. They extend the WS-Agreement standard ducing a new category of terms called Negotiation Terms. They build a

SLA aggregation. They follow BPEL and WS-Policy whereas our form
adheres to WS-Agreement standard.

2.3 Workflow Views

For privacy concerns we will coin the notion of SLA-Views, which is simi
concept of workflow views but is not formally based on it. The concept
flow Views is used to maintain the balance between trust and securit
business partners. Schulz et al. [10] have introduced the concept of vi
cross-organizational workflows and they call it as coalition workflows. C
al. [11] provide a very comprehensive approach that is view based, wel
focused and is applicable to dynamic inter-organizational workflow coo
This means that the cooperation across organizations is described throu
without specifying the internal structure of participating workflows. T
cept of contracts is similar to that of SLA, however, SLAs are more dyn
to negotiation, renegotiation and fault tolerance features. Their is some
vant work done by Chiu et al. [12] interms of a contract model based on
views. They demonstrate how management of contracts can be facilitat
start with an example, highlight domains of different participating orga
and then develop a model to identify the corresponding workflow vie
go on further to develop an e-contract model based on plain text forma
Level Agreements, represented in XML format are more structured an
than the e-contracts. Furthermore their approach starts with defining
an inter-organizational workflow and then describing e-contracts to en
obligatory communication links in the views. Our model allows SLAs to
their individual identity. Therefore, we define views directly on the SL
gation structure rather than on workflows. Moreover, our approach p
formal description of hierarchical SLAs and their aggregation model.

3 Hierarchical Choreography of SLAs

A service level agreement is a contract that defines mutual understand
expectations regarding a service between the service provider and th
consumer. WS-Agreement [13], a standard SLA language from OGF (O
Forum) [14], defines the structure of agreement as depicted in figure 1.
tract should bear an official name. Agreement Context contains inf
about the initiator, the responder and the provider of the agreement; e
time of the agreement; and its template Id. Service Terms define the f
attributes of the agreement whereas the Guarantee Terms contain the n
tional attributes. Guarantee terms further describe the conditions, ser
objectives and business value list related to the agreement. Business
may express the importance of meeting an objective as well as inform
garding penalty or reward.

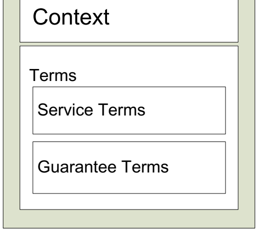


Fig. 1. structure of an agreement in accordance with WS-Agreement spec

Referring to figure 1, we can formally define the Service Terms, and G Terms as part of the encapsulating section Terms.

Definition 1 (Service Term). A service term denoted by $term_s$ is an of the set Service Terms denoted by $STerms$. A $term_s \in STerms$ i such that,

$$term_s = < name, value, type_a >$$

where name and value denote the name and value of a service term a describes its aggregation type.

We have taken the liberty to implant a new mandatory element to Agreement standard, namely, $type_a$. The $type_a$ element corresponds t gregation function that helps us automate the aggregation of SLAs. We its definition to the latter part of the paper where we will discuss the ag process.

Definition 2 (Guarantee Term). A guarantee term denoted by te element of the set Guarantee Terms i.e, $GTerms$. A $term_g \in GTerms$ such that:

$$term_g = < SLO, condition_q, BVL >$$

where SLO represents Service Level Objectives, $condition_q$ represents Q Conditions and BVL represents Business Value List. Combining the a definitions, now we can define the notion Terms in the WS-Agreement

Definition 3 (Term). A $term \in Terms$ is a pair such that

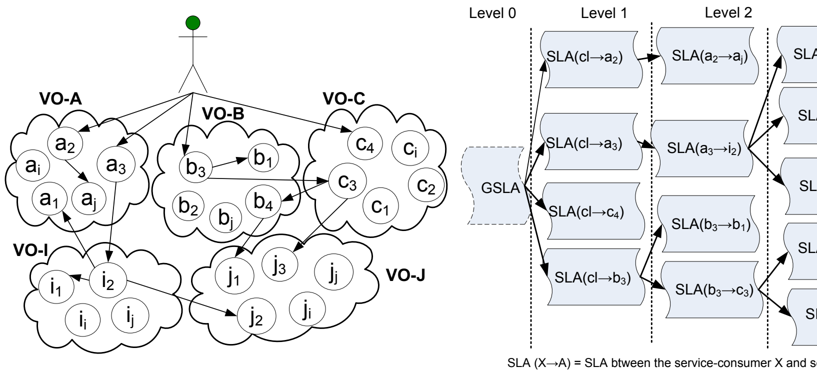
$$term = (term_s, term_g)$$

where $term_s \in STerms$ and $term_g \in GTerms$

Following the above definitions, SLA can now be formally defined as:

Definition 4 (SLA). A service Level Agreement (SLA) denoted by tuple

and Context is a list of strings. Context defines the names of the SLA
the consumer and the initiators. It also contains the duration of the S
parameter *Name* denotes the name of the SLA.



(a) Service Choreography across VOs

(b) SLA-Choreography and SLA depende

Fig. 2. Hierarchical Aggregation of SLAs

A Virtual Organization (VO) in business context, is a temporary c
nent, coalition of geographically dispersed organizations expressing h
mutual trust to collaborate and share their resources and competencies
to fulfill the customers' requests. Web services scattered across variou
istrative domains, when composed together, are said to form service c
phies. In these service choreographies, many service-to-service SLAs ar
The situation becomes even more complex in Business Value Network
services scattered across many such Virtual Organizations (VO) colla
enable complex supply chain networks. One way to visulaize this hiera
terms of dependency layers. Deeper a service in this chain is, more d
its ancestors are. A hierarchy of corresponding SLAs pertains to this
services. There is no multi-level SLA model that can describe the hie
aggregation of SLAs in such Business Value Network. We will call th
chical aggregation of SLAs as SLA-Choreography with relevance to th
Choreography.

In figure 2, we have presented a simplified picture of a cross-VO c
phy. The client (that may be a workflow process) is directly connected
services, scattered across three VOs: VO-A, VO-B, VO-C. These ser
coordinating with other services to carry out their jobs. This coordin
sults into service chains, distributed across multiple Virtual Organisati
scenario can be compared with a simple Business Value Network. Th
services play the producer-consumer roles in this service choreograp

organization of SLAs. There may be several dependency layers in t
Choreography. The dependency increases along the hierarchy. The ag
effect of this dependency travels from the very bottom towards the
This SLA aggregation is depicted in Fig. 2. In this hierarchy the SLA
are connected to the client process, are said to exist on level 1. This
indicates a supply chain type of correspondence among the services. Th
also denote the visibility levels of service providers and the client. The
concern only with the services immediately connected to it and can
beyond. Similarly a service can see its coordinating services i.e its
and its consumers with which it is making service level agreements.
information about the rest of the service choreography. Despite of it
concerns, a service is dependent on its lower services. The effect of SLA
among the services at lower levels is bubbled up through the upper la

There are many interesting questions that need answers: What tru
will bind together the Business Value Networks? Who will manage t
Choreography? How to monitor and validate this SLA-Choreography?
these questions are related to our overall research agenda but are be
scope of this paper. In this paper we focus on an even more basic pro
develop a formal model that can describe this SLA-Choreography and
an aggregation model for hierarchical SLAs while protecting the privacy
of the stakeholders at the same time. For this purpose, we introduce th
of SLA-Views.

4 SLA Views

The concept of *Views* originates from the field of databases and has been
fully adapted in business workflows [11][5]. In workflows, a view can be
of that workflow or can be a representation of that workflow in aggr
abstracted fashion. We have also employed the notion of views to re
subset of SLA-Choreography. As the matter of fact the notion of SLA
related to that of workflow views in a very general sense. In formal se
Views are absolutely different from the workflow views. SLA-Choreo
not a workflow so the rules of workflows are not applicable on it. For in
a workflow, rules such as: there should be a single start and single exit
split should have a join, do not apply on SLA choreography.

A view in an SLA-Choreography represents the visibility of a busines
Every service provider is limited only to its own view. A partner (for
a service) makes two kinds of SLAs: the SLAs for which it acts as a
and the SLAs for which it is a provider. For clarity, we name these two
the consumer-oriented SLAs and the producer-oriented SLAs respecti

In figure 3, SLAs are connected to small circles, which we call *ag*
points, by certain edges called *dependencies*. There are two types of depe

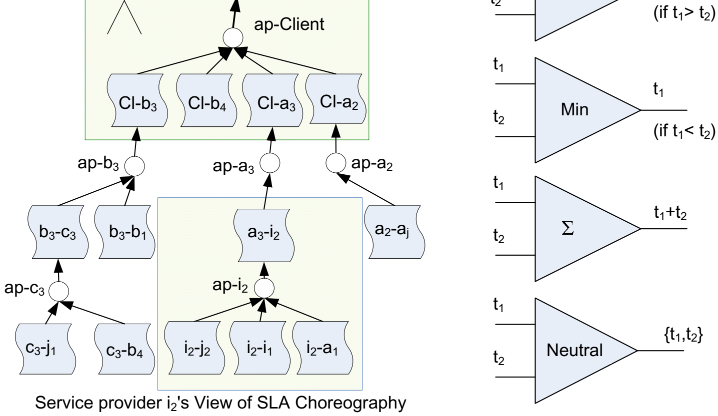


Fig. 3. Different Views in the SLA-Choreography And Some Basic A
Function

Consumer-oriented SLAs are connected to the aggregation points fr
by the *sink dependencies* and the producer-oriented SLAs are conne
above by *the source dependencies*. To understand the overall picture of
Choreography, we need to formalize these concepts.

Definition 5 (Aggregation Point). An Aggregation Point *ap* is
such that

$$ap =< aggsla >$$

where *aggsla* is the aggregated SLA produced by aggregating the c
oriented SLAs connected to it. In figure 3, $ap\text{-}i_2$ is an aggregation
aggregation point is the point where the consumer-oriented SLAs (of
sumer service) are aggregated and on the basis of their aggregated co
service is able to decide what it can offer as a provider. The master-slave
ships in Business Value Networks are directly translated to producer-
model with one service provider (Enterprise) as a producer and other
sumer. So both the producer and the consumer enterprises will have
aggregation points connected together through their mutual SLA. Ho
peer-to-peer relationships, both peers act as producer and consumer o
This issue can be easily resolved by translating peer-to-peer relation
producer-consumer model. For this purpose, we device the concept of v
gregation point (vap) to automate the aggregation process. Virtual ag
point is discussed in detail in section 6.

Now let us define dependencies which have been shown in figure 3(a
joining the aggregation point with the producer and consumer orient
The Aggregation Point $ap\text{-}i_2$ is connected with three consumer-orien
and one producer-oriented SLA through dependencies.

where ap is the aggregation point and sla is the producer-oriented SLA.
3(a), it is represented by the directed edge from the aggregation poin
the producer-oriented SLA, $sla_{a_3-i_2}$.

Each $dep_{src} \in Dep_{src}$, where Dep_{src} is the set of all source dependenci
the SLA-Choreography. Let

$$source : (ap) \rightarrow dep_{src}$$

$source(ap_i)$ is the unique $s \in Dep_{src}$, for which a unique producer
SLA exists with $s = (ap_i, sla_i)$. This means that the function source m
aggregation point ap_i to a unique SLA through a unique source depen

Definition 7 (Sink Dependency). A sink dependency dep_{sink} is a

$$dep_{sink} =< sla, ap >$$

where ap is the aggregation point and sla is the consumer-oriented
Figure 3, it is represented by the directed edge from the consumer-orie
i_2-i_1 to the aggregation point ap-i_2. The aggregation point ap-i_2 is c
with three sink dependencies.

Each $dep_{sink} \in Dep_{sink}$, where Dep_{sink} is the set of all sink dependenci
the SLA-Choreography. Let

$$sink : (ap) \rightarrow P(dep_{src})$$

where $P(Dep_{sink})$ is the power set of Dep_{sink}.

$sinks(ap_i)$ is the set $S_{sink} \in P(Dep_{sink})$, i.e. $S_{sink} \subseteq Dep_{sink}$ such
each $s_i \in S_{sink}$ a unique consumer oriented SLA exists with $s_i = ($
This means that the function *sinks* maps a set of consumer-orieted S
unique aggregation point such that each consumer-oriented SLA sla_i i
through a unique sink dependency s_i.

Definition 8 (Dependency). A dependency Dep is a set that is the
two sets namely Dep_{src} and Dep_{sink} which are pairwise disjoint, i.e.

$$Dep = Dep_{src} \cup Dep_{sink}$$

$$Dep_{src} \cap Dep_{sink} = \phi$$

Based on these definitions, in figure 3, we see that the producer-orie
(a_3-i_2) is dependent on the terms of the corresponding consumer-orient
aggregated at ap-i_2 . For example the bandwidth and space aggregate
would be the upper limit of what service i_2 can offer to service a_3. At
time service i_2 will have to decide about its profit on the basis of the in
about total cost in the aggregated SLA. The aggregation point in thi
also a decision point for a service.

that

$$slaview_i =< sla_p, dep_{sr}, ap_i, SLA_c, Dep_{sn} >$$

where sla_p = producer-oriented SLA, SLA_c= Set of consumer-orient
dep_{sr}= source dependency, Dep_{sn}= set of sink dependencies, and ap_i
gation point. Each aggregation point ap_i in the SLA-Choreography co
to a unique *sla-view*$_i$.

In figure 3, the SLA-Views of the client and a service are highlighted.

Definition 10 (SLA-Choreography). An SLA_{chor} is a tuple such t

$$SLA_{chor} =< SLA, APoints, Deps >$$

where SLA is set of all *sla* within an SLA-Choreography, $APoints$
aggregation points ap, and $Deps$ is set of dependencies dep. Anothe
describe the SLA-Choreography is in terms of SLA-Views, i.e.

$$SLA_{chor} = \cup_{i=1}^{n} slaview_i$$

This means that the whole SLA-Choreography may be seen as an in
of several SLA-Views. In terms of Business Value Networks, it should
that SLA-View defines boundaries of a stakeholder. The aggregatio
is performed at every aggregation point. Each aggregation point, w
denotes a dependency level, belongs to one of the service providers.
each service provider is limited to its own aggregation information,
information is in fact dependent on the aggregation information at lo
The sustainability of this business network requires all the stakeholder
each other and their ability to maintain their privacy at the same ti
Views maintain a balance between this privacy and trust.

5 Aggregation Process

In the aggregation process, terms of the consumer-oriented SLAs a
gated. WS-agreement has no direct support for such an aggregation
troduced an attribute for aggregation type namely, "$type_a$" in the Def
WS-Agreement gives the liberty to incorporate *any* external schema.
$type_a$ can be made an essential part of the service terms and will des
the corresponding service will behave during the aggregation process
define $type_a$ in a formal way, as follows:

Definition 11 (The function type$_a$). A $type_a \in Types$ is a func
maps a set of tuples to a single tuple which is the aggregation of that

$$type_a : tuples(term) \rightarrow term$$

term. Its result is *aggsla* in the aggregation point (please see Definition
term in *aggsla* is computed by applying the type function for that
the values of the terms for all the dependent (consumer-oriented) SL
define that term. In the present context, we define four types of term
sumtype, maxtype, mintype and neutral but new types can be added
to the situation, i.e.

$$Types = \{sumtype, maxtype, mintype, neutral\}$$

These functions have been depicted in figure 3(b). The function sum
be formally defined as follows.

Definition 11.1 (The function sumtype)

$$sumtype \in Types(\Leftrightarrow sumtype : tuples(term) \rightarrow term$$

$$sumtype(term_1, ...term_n) = \sum_{i=1}^{n} term_i$$

$type_a$ is an aggregation function that aggregates n number of terms
term. sumtype is of the type of $type_a$ and takes the summation of
Examples include terms for storage space, memory, availability and co

Definition 11.2 (The function maxtype)

$$maxtype \in Types(\Leftrightarrow maxtype : tuples(term) \rightarrow term$$

$$maxtype(term_1, ...term_n) = \max_{i=1}^{n} term_i$$

maxtype is an aggregation function that aggregates n number of terms
term. It does so by picking up the maximum of these terms which repre
aggregation of all the input terms.If several terms addressing the sa
are being aggregated and their type has been declared as maxtype
the term pertaining to the maximum value will become part of the a
SLA. Examples include latency, which may become a bottle neck for
process and an activity with highest latency will directly contribute
in negative sense) to the throughput of a workflow sequence.

Definition 11.3 (The function mintype)

$$mintype \in Types(\Leftrightarrow mintype : tuples(term) \rightarrow term$$

$$mintype(term_1, ...term_n) = \min_{i=1}^{n} term_i$$

mintype is an aggregation function that aggregates n number of terms
term. It does so by picking up the minimum of these terms which r
the aggregation of all the input terms. Similar to maxtype, when seve
addressing alike utilities are being aggregated and their type has been

bottleneck for the whole sequence making other activities with higher b
ineffective.

Definition 11.4 (The function neutral)

$$neutral \in Types(\Leftrightarrow neutral : (term) \rightarrow term$$

$$neutral(term_i) = \ term_i$$

neutral is an aggregation function that includes all the input terms s
without any processing. This function is applied on those terms whic
be mixed with other terms and need to preserved in the aggregatio
as separate terms.The terms declared as neutral are unaffected throug
gregation process and are just copied in the aggregated SLA. They
services which are independent from similar services, for example i
some valuable data in a certain organization or discount in a specific s

So far we have defined only four types of terms but it is important
that this enumeration can be extended without affecting the generic
of the $type_a$ function. In certain cases, for example calculating the r
penalty expressions, logical operations will also be required. On sim
we can define logical functions such as AND, OR, XOR to integrate t
level objectives or other constituents of Guarantee Terms to form
aggregation expressions.

6 A Case for Hierarchical Aggregation of SLAs in Business Applications

NESSI, in their Grand Vision and Strategic Research Agenda (SRA) [
Value Networks as the ways in which organisations interact with e
to drive increased business value. Figure 4 shows their example Busin
Network (BVN) where the Enterprises A and D have been shown to c
on the development of a new product. Enterprise A has subcontractors
whereas the enterprise has E and F as subcontractors. The Enterprises
form a peer-to-peer relationship between themselves.

So far, we have discussed the aggregation of SLAs in context with
position of services in a producer-consumer manner, along service val
This service level SLA aggregation model can be scaled up to enterp
It can conveniently describe both master-slave and peer-to-peer rel
in Business Value Networks. Master-slave relationship can be simpl
on the producer-consumer model where an SLA is formed between t
provider and the client. However, in peer-to-peer relationships, the par
enterprises are acting as the service provider and the client at the same
form a WS-Agreement compliant SLA between them, one party can

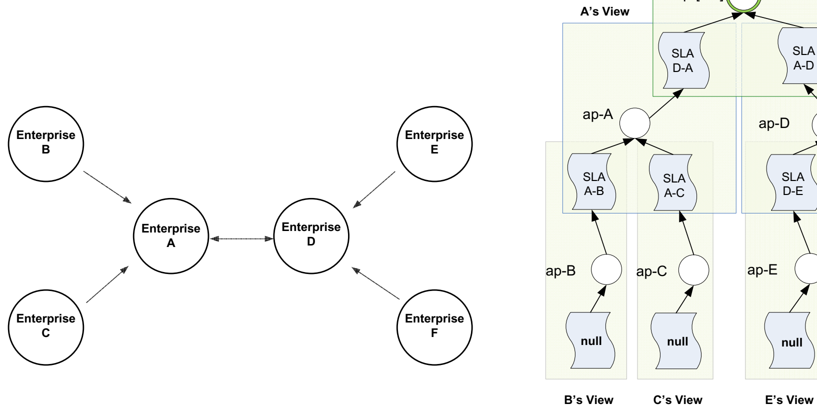


Fig. 4. A Business Value Network and its corresponding SLA Choreography
ferent Enterprises' Views

treated as a service provider or a service consumer in context with som
Therefore a peer-to-peer relationship needs to be dissolved into two p
consumer relationships with a separate SLA associated with each of th
we would like to define a Virtual Enterprise Organisation (VEO). Acc
NESSI's definition [2] VEOs are formed when two or more administr
mains (and hence their Enterprise Grids) overlap and share resources.
describes that the reality of VEO is that only a subset of the overall Gr
an enterprise is likely to be contributed to this virtual organisation. The
ing relationships among different enterprises within a VEO can be ma
or peer-to-peer or a combination of both. We will apply the concept o
peer-to-peer relationships in figure 4. If we consider the enterprises A
form a Virtual Enterprise Organisations (VEO), their SLAs are aggreg
virtual aggregation point (vap) that represents this VEO. The virtual
tion point is important to be represented because it in turn describes
view of the resulting VEO which is different from the SLA views of A an
shared functionality of the VEO is described in the aggregated SLA
within the vap-[AD]. Note that the big brackets have been adopted to
the jointly contained capabilities of enterprises A and D. The terms o
are aggregated through aggregation functions described in section 5. T
marked as neutral are not merged and kept separate in the aggregated S
virtual aggregation point also denotes the decision point of the resulting
policies such as distribution of revenue and cost of offered services wi
decided inside it. From a practical perspective, there are numerous is
as trust, security, heterogeneity related to SLA aggregation among pee

prises [2] can be easily described through our model. The concept of i
or cloud of clouds [16] is becoming very popular these days, which re
virtual collaboration of clouds. Such a virtual collaboration among clo
straightforwardly on our SLA aggregation model.

7 Motivational Scenario

In the following section, we will present a motivational scenario of a business value network which is enabled by the aggregation mechanism above. Arfa is visiting ULM. She is shooting movies and capturing snaps the camera, built in her mobile phone. The mobile device has limite space but luckily she knows a web service that can archive, enhance her movies online as soon as she completes a recording. She is also v excited to share her experiences with her family and friends. Therefore to update some blogs with images of the places and their historical de Her friend told her about an online service that can collect images fro phone, print them and send them as postcards. So, she would like to tasks: automatically store and host her movies to external storage fr she and her friends can watch anytime using their mobile or stati automatically print some selected images as postcards and mail the family and friends through regular post; update some blogs with im their historical descriptions. The SLA-Choreography resulting from th workflow is shown in figure 5. There are two services, namely the h service and post-photo service. The host video service downloads the vi the mobile device, enhance s it and archives it. Any authenticated play the video in a youtube like style. The Post-Photo service ma with two services: the Print&Post service and E-Post service. E-Po is able to do its task by contracting two service namely Blog-Service a Service. The Blog service can automatically update the blogs with t and automatically generate stories about their historical significance on of their exact address. MMS service sends the selected images to friend mobile phones.

The SLA-Choreography resulting from this scenario is depicted in We can see the aggregation functions described in figure 3(b) being a the scenario shown in figure 5. It is evident that the resolution offered Video service is the minimum of the three services below it. So at the ag point ap-S_1, the aggregation function Min will choose only minimum of resolutions as their aggregation types have been declared “min”. On grounds, the job completion time for E-Post service is the maximum o Blog service and MMS service beause it is of “maxtype”. The total cos client has to pay is the sum of the cost incurred on Host-Video servic cost spent on Post-Photo service because cost has been declared as “s

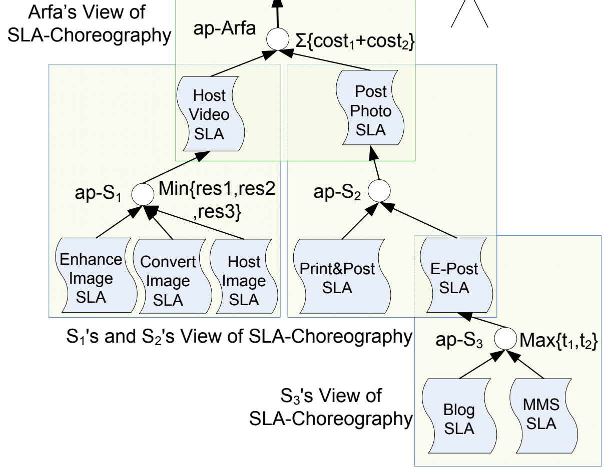


Fig. 5. Different Partners' SLA Views in Motivational Scenario

We take the liberty of importing external schema into WS-Agreement Description Terms' section. The following chunk of Schema allows this

```
<xs:complexType name="ServiceDescriptionTermType">
   <xs:complexContent>
      <xs:extension base="wsag:ServiceTermType">
         <xs:sequence>
            <xs:any namespace="##other" processContents="st
         </xs:sequence>
      </xs:extension>
   </xs:complexContent>
</xs:complexType>
```

The above schema enables us to include an XML structure of elements to any external Schema. This makes it possible to incorporate the ag type (typea) element inside a Service Description Term. A simple s accomplish this can be written as follows.

```
<?xml version="1.0" encoding="utf-16"?> <xs:schema
xmlns:myns="http://schemas.xyz.com" xmlns="http://www.mynamespace
targetNamespace="http://www.mynamespace.com"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:simpleType name="aggregationType">
 <restriction base="xs:string">
   <enumeration value="Mintype"/>
   <enumeration value="Maxtype"/>
   <enumeration value="sumtype"/>
```

```
  <xs:element name="Resolution">
    <xs:complexType>
      <xs:sequence>
        <xs:complexType name="ResolutionXY">
          <xs:sequence>
            <xs:element name="ResolutionX" type="xs:integer"/>
            <xs:element name="ResolutionY" type="xs:integer"/>
          </xs:sequence>
        <xs:element name="aggregationType" type="xs:aggregationTy
       </xs:complexType">
      </xs:sequence>
    </xs:complexType>
  </xs:element>
... </xs:schema>
```

Then the service Description Term namely "resolution" for the Enhan
service may be expressed as follows.

```
<wsag:ServiceDescriptionTerm wsag:Name=Resolution"
wsag:ServiceName="Enhance-Video">
   <myns:ResolutionXY>
      <myns:ResolutionX> 640</myns:ResolutionX>
      <myns:ResolutionY>480</myns:ResolutionY>
   </myns:ResolutionXY>
   <myns:aggregationType> mintype</myns:aggregationType>
</wsag:ServiceDescriptionTerm>
```

The aggregationType (i.e. $type_a$) declares Resolution as a minType ter
it will be aggregated with other minType terms, only the minimum of th
will become part of the aggregated SLA. Other aggregation types list
schema can be expressed and aggregated in a similar fashion.

8 Conclusion

We presented a view based formal model to describe hierarchical Serv
Agreements in supply chain scenarios such as Business Value Networ
Views help to maintain balance between trust and privacy. Our model
basic aggregation constructs that are used in the aggregation of SLAs. T
aggregation process stays in compliance with the WS-Agreement stand
to the limited scope of this paper we could not include various deta
research related to different aspects of Business Value Networks such
of value and business models. However, We plan to address these
context with the Cloud Computing, as a separate research paper. I
we will continue our work on implementing a secure aggregation and
framework for SLAs in heterogeneous Virtual Organizations.

lines regarding the application areas of our research and thus helped u
duce a much improved Camera Ready Version of our paper. This v
partly supported by the project grant number IP395009, funded by U
of Vienna.

References

1. Blake, M.B., Cunnings, D.J.: Workflow composition of service level agree International Conference on Services Computing, SCC 2007 (2007)
2. NESSI-Grid, http://www.soi-nwg.org/doku.php?id=sra:description cess: March 12, 2009)
3. Project, S.: (March 12, 2009), http://www.sla-at-soi.org/index.html
4. Liu, D.R., Shen, M.: Workflow modeling for virtual processes: an order-p process-view approach. Information Systems 28, 505–532 (2002)
5. Liu, D.R., Shen, M.: Business-to-business workflow interoperation process-views. Decision Support Systems 38, 399–419 (2004)
6. Eder, J., Tahamatan, A.: Temporal consistency of view based interorga workflows. In: 2nd International United Information Systems Conferenc (2008)
7. Frankova, G.: Service level agreements: Web services and security, pp. Springer, Heidelberg (2007)
8. Unger, T., Leyman, F., Mauchart, S., Scheibler, T.: Aggregation of se agreement in the context of business processes. In: Enterprise Distribut Computing Conference (EDOC 2008), Munich, Germany (2008)
9. Aiello, M., Frankova, G., Malfatti, D.: What's in an agreement?An an an extension of WS-agreement. In: Benatallah, B., Casati, F., Traverso ICSOC 2005. LNCS, vol. 3826, pp. 424–436. Springer, Heidelberg (2005
10. Schulz, K.A., Orlowska, M.E.: Facilitating cross-organisational workflo workflow view approache. Data and Knowledge Engineering 51, 109–147
11. Chebbi, I., Dustdar, S., Tata, S.: The view based approach to dyna organizational workflow cooperation. Data and Knowledge Engineering 5 (2006)
12. Chiu, D., Li, K.K.Q., Kafeza, E.: Workflow view based e-contracts i organisational e-services environment. Distributed and Parallel Data 193–216 (2002)
13. Ludwig et al: Web service agreement (ws-agreement). gfd.107 proposed r dation (last access: July 12, 2008)
14. (OGF), O.G.F.: http://www.ogf.org/ (last access: March 12, 2009)
15. ul haq, I., Huqqani, A.A., Schikuta, E.: A conceptual model for aggreg validation of slas in business value networks. In: The 3rd International C on Adaptive Business Information Systems, ABIS 2009 (2009)
16. Jha, S., Merzky, A., Fox, G.: Using clouds to provide grids with highe abstraction and explicit support for usage modes. Concurrency and Con Practice and Experience 21(8), 2087–1108 (2009)