A Scenario-based Framework for Perceiving Crossbreed Service Interactions

Natalia Kryvinska*, Christine Strauss

*Department of eBusiness, School of Business, Economics and Statistics, University of Vienna, Bruenner Strasse 72, A-1210 Vienna, Austria

Abstract

The service-oriented computing provides technologies allowing multiple enterprises to integrate their businesses over the Internet. Typical execution behavior in this type of converged schemes comprises a network of autonomous peers interacting with each other. Modeling and analyzing the interactions among different services is a crucial problem in this domain. It is a particularly challenging task since no single party has access to the internal states of all the participants. Desired behaviors have to be specified as constraints on the interactions among different peers since the interactions are the only observable global behavior. Besides, it might be meaningful to specify the interactions among different parties before the services are implemented. Undeniably, one of the main requirements is preserving the autonomy of each participating partner during the interaction, without restricting the overall goals of the common process. Thus, mechanisms orchestrating distributed service workflows are needed. The scenario-based framework for perceiving crossbreed service interactions examined in this paper efforts to meet these requirements.

© 2011 Published by Elsevier Ltd. Selection and/or peer-review under responsibility of [MobiWIS-2011]

Keywords: Crossbreed Service Interactions, Hybrid/Converged Services; Services Orchestration and Choreography; Web Services

1. Introduction

Modern crossbreed, integrated network technologies allow developing new collaboration business paradigms, such as virtual enterprises, where different companies pool together their resources to offer more complex, added-value products and services. Besides, network technologies and Internet make services easily accessible and thus they allow composing virtual enterprises in very flexible ways [1, 2, 23].

The term integrated or converged network has been used primarily as a reference to integration of the traditional telephone networks with IP-based networks. However, lately the convergence has been interpreted as the more general integration of wireline and wireless networks, also referred to as fixed-mobile convergence (FMC). To merge those views, it could be stated that a converged network is a communications network that seamlessly integrates circuit, packet, wired, and wireless networks. This characterization allows moving further on to services exploring. Without getting into a rigorous definition of a service, since this may differ from one context to another,
it is a fact that each network layer in any communications network offers some services to the layer above, until eventually they are composed into an end-to-end service that is offered to an end consumer (e.g., an end user or a system operating on behalf of an end user or business entity). Among other reasons, network convergence is desirable because of the potential to offer better end-to-end services. But, at the same time, service convergence cannot be reduced to services offered by converged networks simply because users expect more than just removing the technical differences between networks; users expect composed services that satisfy their increased needs for collaboration, customization, and personalization. A battle that is going on between network equipment providers and IT vendors also adds a new convergence dimension - convergence between architectures and technologies from different domains, all with the declared intent to improve the end-to-end services to the consumer [3 ÷ 7].

Taking the above described pros and cons into consideration while trying to keep the definition reasonably simple, it is anticipated [9] that converged services are complex services that combine different media, data communications, and telco services through seamless interactions of wireless/wireline technologies, features customized by policies, and personal preferences to deliver new value-added services. It seems reasonable to assume that converged services are services that may be offered when making use of converged networks (as defined), although that may not be the only possibility [3÷7, 8, 9].

The service-oriented computing provides technologies allowing multiple enterprises to integrate their businesses over the Internet. Typical execution behavior in this type of converged schemes comprises a network of autonomous peers interacting with each other. Modeling and analyzing the interactions among different services is a crucial problem in this domain. It is a particularly challenging task since no single party has access to the internal states of all the participants. Desired behaviors have to be specified as constraints on the interactions among different peers since the interactions are the only observable global behavior. Besides, it might be meaningful to specify the interactions among different parties before the services are implemented. Undeniably, one of the main requirements is preserving the autonomy of each participating partner during the interaction, without restricting the efforts to reach the overall goals of the common process. Thus, mechanisms orchestrating distributed service workflows are needed. The scenario-based framework for perceiving crossbreed service interactions examined in this paper efforts to meet the requirement [10 ÷ 12, 23].

2. Behavioral Models for Network Services Orchestration

The orchestration of the network services refers to the automated arrangement, coordination and management of computer systems, storage, security and networks in order to efficiently deliver application services to end users. Orchestration triggers an executable process that involves centrally controlled message exchanges among network entities. As part of the orchestration of network services, the orchestration system provides a layer of abstraction between the application services and the infrastructure. This layer of abstraction is sometimes referred to as network services virtualization.

In conceptual model of an orchestration system, polices define the relationship between users, computing resources, security and network services. These policies are automatically translated in real-time into device configurations that dynamically provide the necessary resources and/or modify the resource pool to bring it into alignment with the service definition. User access information is updated in user directories and network access control system databases. The virtual server requirements of the service are communicated to the hypervisor management system. The hypervisor management system provides the required services. The hypervisor manager will also be able to make the necessary configuration changes for virtual switches and virtual appliances under its control.

Network elements not under the control of the hypervisor management system (e.g., routers, switches, firewalls, and other external devices that provide secure access to networked services) are controlled on an individual basis by the orchestration system. The orchestration system also communicates with traditional network management solutions, such as fault and performance management systems, which help to monitor services and assure that type of service level specified in the policies delivered for each network service [8, 13].

Service choreography, in context of the conceptual model, is a form of service composition in which the interaction protocol between several partner services is defined from a global perspective [14].
Thus, during run-time each participant in service the service value chain executes its part (i.e., its role) according to the other participants’ behavior [15]. The choreography’s role identifies the expected messaging behavior of the participants in terms of sequencing and timing of the messages that they may consume and produce [16].

Service choreography is better understood through the comparison with another paradigm of service composition: i.e. service orchestration. Whereas, in service choreographies the logic of the message-based interactions among the participants is specified from a global perspective. In service orchestration, however, the logic is specified from the local point of view of one single participant, called the orchestrator. In the service orchestration language BPEL, for example, the specification of the service orchestration (e.g., the BPEL process file) can be deployed on the service infrastructure (for example a BPEL execution engine like Apache ODE). The deployment of the service orchestration specification creates the composed service.

Service choreographies are not executed: they are enacted. Service choreography is enacted when its participants execute their roles [17]. That is, unlike service orchestration, service choreographies are not run by some engine on the service infrastructure, but they “happen” when their roles are executed. This is because the logic of the service choreography is specified from a global viewpoint, and thus it is not even realized by one single service like in service orchestration [18].

A choreography model describes collaboration processes between collections of services to achieve a common goal. It captures the interactions in which the participating services engage to achieve this goal and the dependencies between these interactions, including: causal and/or control-flow dependencies (i.e., a given interaction must occur before another one, or an interaction triggers another one), exclusion dependencies (a given interaction excludes or replaces another one), data-flow dependencies, interaction correlation, time constraints, transactional dependencies, etc. The choreography does not describe any internal action of a participating service that does not directly result in an externally visible effect, such as an internal computation or data transformation. Choreography captures interactions from a global perspective meaning that all participating services are treated equally. In other words, choreography encompasses all interactions between the participating services that are relevant for the choreography’s goal [19, 20].

Thus, the service choreography identifies the set of allowable conversations for composite services. An orchestration, on the other hand, is an executable specification that identifies the steps of execution for the enterprises [10 – 12].

In conformity with the above definitions, we examine in the next section scenarios of an “Internet call waiting” service orchestration developed from multiple interconnected perceptions.

3. An Example of Service Scenario Orchestration Urbanized from Multiple Interconnected Perceptions

3.1. Internet call waiting

Internet call waiting (ICW) is a service that enables a user engaged in a dial-up Internet session to be alerted when an incoming call has arrived. After the Internet user has been alerted, he is given several options for handling the call (e.g., forwarding it, sending a waiting announce/tone, accepting the call over telephone network suspending the Internet session, or accepting the call over IP keeping alive the Internet session). In parallel the caller is announced that the callee is busy and (s)he is asked to hold the line. In order to enable the service the subscriber has to use a client software that performs the registration phase by storing the association between the telephone network line number and the IP address of his/her Internet session.

The service features those have an impact on the context, e.g., handling an incoming call, are as follows - the subscriber receives a telephone call directed to his phone, which is busy, since it is engaged in a dial-up Internet session. The IN service switching point (SSP) triggers the service logic to handle this call event. The caller is connected to an intelligent peripheral in order to send a message to inform other side to wait (call queuing). The service retrieves the IP address of the callee and then, depending on the user profile, notifies the incoming call to the Internet user with caller number or name or other call relation information. The Internet user may choose different options:

(a) accept call on PC using voice over IP;
(b) accept call on phone;
(c) suspend IP session and answer the call on the phone;
(d) reply the caller with a pre-registered message;
(e) reject the call [3, 5, 6, 7, 21, 22].

3.2. ICW - Connection to Internet

The ICW subscriber connects to Internet by means of a dial-up connection (Table 1, Fig. 1) [3 ÷ 7, 21, 22].

![Fig. 1. ICW – Connection to Internet](image1.png)

![Fig. 2. ICW – Incoming Call Notification](image2.png)

Table 1. ICW – Connection to Internet

<table>
<thead>
<tr>
<th>Pre-conditions</th>
<th>User A has subscribed to ICW service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action 1</td>
<td>The dial-up connection is established.</td>
</tr>
<tr>
<td>Action 2</td>
<td>User A launches the Internet connection software, which dials the ISP phone number, that is an number which represents an IN service. As a result of the pre-arranged agreement between the Internet service provider and the network provider the DP3 (AnalyzedInformation) was set. The SSP triggers the service logic by sending an InitialDP message to the IN service control point (SCP). Then the SSP, through a gateway, notifies the ICW service logic about a network event related to an incoming call for the ISP phone number. The service logic is executed within a kind of SLEE (service logic execution environment) that provide API to interact with network functionality.</td>
</tr>
<tr>
<td>Action 3</td>
<td>The ICW service logic consequently sets TDP13 using some management interface on the SSP. The ICW service logic then subscribes to call event related to user A call link disconnection since it needs to disarm the TDP13, when the Internet connection is disconnected.</td>
</tr>
<tr>
<td>Action 4</td>
<td>The ICW service logic gives instruction to the network to route the call to the address of the network access server (NAS) to be connected. Furthermore it arms DP7 (OAnswer) in order to be notified about the result of the call routing. The connection is set up between the SSP and the NAS and consequently the network acknowledges the establishment of the connection. Since the ICW service logic needs to monitor NAS disconnection as well as user A disconnection, it subscribed to call event (DP9ODisconnect) related to NAS disconnection (this subscription could not be requested before the successful notification of the call routing to the NAS).</td>
</tr>
<tr>
<td>Action 5</td>
<td>When the network acknowledges the successful establishment of the call link towards the NAS, the call processing is stopped at the DP7 (OAnswer) since it has been set as an EDP-R. The ICW service logic instructs the SCP to continue the call processing (this is mapped onto the INAP operation Continue). The subscriber is now requested to authenticate through a login and password. This subscriber data are sent to a RADIUS server to authenticate him/her and to retrieve from a directory server user’s A profile based on its home phone number. The association between the IP address of user A and home phone number of user A is stored in his user profile.</td>
</tr>
<tr>
<td>Action 6</td>
<td>The software in charge of handling invitations is launched automatically after the connection to Internet is established e.g. IETF SIP UAC.</td>
</tr>
</tbody>
</table>
Trigger detection point (DP) 13 (in ETSI core INAP terminology DP13 corresponds to TCalledPartyBusy event) is armed so that when there is an incoming call for user A and his line is busy the ICW is notified.

### 3.3. ICW - Incoming Call Notification

The user B tries to call user A at his home phone number (Table 2, Fig. 2) [3 ÷ 7, 21, 22].

**Table 2. ICW – Incoming Call Notification**

<table>
<thead>
<tr>
<th>Pre-conditions</th>
<th>User A is connected to Internet.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action 1</strong></td>
<td>User B dial user’s A phone number.</td>
</tr>
<tr>
<td><strong>Action 2, 3, 4</strong></td>
<td>User’s A SSP detects that user’s A phone line is busy and according to the previous arming of TDP13, an InitialDP message is sent to the SCP in order to notify that a call has been triggered.</td>
</tr>
<tr>
<td><strong>Action 5</strong></td>
<td>The call is routed to an interactive voice response (IVR), which implements IN special resource functions.</td>
</tr>
<tr>
<td><strong>Action 6</strong></td>
<td>The ICW service logic instructs to send a ConnectToResource message to the SSP. The SSP then establish the connection to the IVR.</td>
</tr>
<tr>
<td><strong>Action 7</strong></td>
<td>The ICW service logic sends an invitation (e.g. SIP INVITE message) to user A with information about the caller (e.g. calling line identity) asking him either to:</td>
</tr>
<tr>
<td></td>
<td>• accept call on PC</td>
</tr>
<tr>
<td></td>
<td>• accept call on phone</td>
</tr>
<tr>
<td></td>
<td>• suspend IP session and answer the call on the phone</td>
</tr>
<tr>
<td></td>
<td>• reply the caller with a pre-registered message</td>
</tr>
<tr>
<td></td>
<td>• reject the call user’s A choice is sent back to the ICW service logic via a SIP response message.</td>
</tr>
<tr>
<td><strong>Action 8</strong></td>
<td>The IVR reports the end of the announcement to the SSP.</td>
</tr>
<tr>
<td><strong>Post-conditions</strong></td>
<td>User A is notified of an incoming call and has to choose either to reject the call, to answer the call over a PSTN line or to answer the call over IP.</td>
</tr>
</tbody>
</table>

### 3.4. ICW - Rejecting a Call

The user A is invited to a call by a user B and chooses to reject it (Table 3, Fig. 3) [3 ÷ 7, 21, 22].

**Table 3. ICW – Rejecting the Incoming Call**

<table>
<thead>
<tr>
<th>Pre-conditions</th>
<th>User A is connected to Internet via a dial-up connection, an incoming call has arrived for User A, User A chooses to reject the call.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Action 1</strong></td>
<td>An user B invites the user A to a call.</td>
</tr>
<tr>
<td><strong>Action 2, 3, 4</strong></td>
<td>The user A chooses to reject it.</td>
</tr>
<tr>
<td><strong>Action 5, 6</strong></td>
<td>When the Internet user rejects the incoming call the ICW service logic requests to play an announcement to the user B saying that the callee has rejected the call.</td>
</tr>
<tr>
<td><strong>Action 7</strong></td>
<td>The service logic asks to play an announcement.</td>
</tr>
<tr>
<td><strong>Action 8</strong></td>
<td>When the announcement is finished the SSP notify the ICW service logic about the end of it.</td>
</tr>
<tr>
<td><strong>Post-conditions</strong></td>
<td>The ICW service logic then releases the call initiated by the user B.</td>
</tr>
</tbody>
</table>

The incoming call is terminated. The dial-up connection is still active.
3.5. ICW - Accepting a Call On the Phone

The user A is invited to a call by a user B and chooses to answer on his PSTN line (Table 4, Fig. 4) [3 ÷ 7, 21, 22].

Table 4. ICW – Accepting the Call on the Phone

<table>
<thead>
<tr>
<th>Pre-conditions</th>
<th>Action 1</th>
<th>Action 2, 3</th>
<th>Action 4</th>
<th>Action 5</th>
<th>Action 6</th>
<th>Action 7</th>
<th>Action 8</th>
<th>Action 9</th>
<th>Action 10</th>
<th>Action 11</th>
<th>Post-conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>User A is connected to Internet by means of a dial-up connection, an incoming call has arrived for the User A, the User A chose to answer the call on his PSTN line.</td>
<td>User A is invited to a call by a user B.</td>
<td>The ICW client software disconnects the dial-up connection.</td>
<td>The “disconnect” signal is sent to the SSP. This is triggered by the SSP since an EDP9 (ODisconnect) related to the call between user A and the NAS was previously armed.</td>
<td>Consequently the SSP sends an EventReportBCSM operation to the ICW service logic.</td>
<td>Since a connection to an IVR was still established, the ICW service logic releases this connection. The SSP disconnect the connection to the IVR.</td>
<td>And, sends to the NAS a message to disconnect it.</td>
<td>As the result of the previous arming of an EDP9 (ODisconnect) related to the disconnection of the NAS the SSP sends an EventReportBCSM operation to the ICW service logic.</td>
<td>The NAS detects the end of the dial-up connection and instructs the RADIUS gateway to stop the accounting.</td>
<td>The RADIUS gateway asks the accounting server to stop the accounting for user’s A account.</td>
<td>The ICW service logic has just released the call to the NAS, therefore it tries to route the call to user’s A phone line.</td>
<td>The dial-up connection is terminated and the call is established between the User A and the User B using PSTN to PSTN connection.</td>
</tr>
</tbody>
</table>

3.6. ICW - Accepting The Call On IP

The user A during his Internet session chooses to answer an incoming call using VoIP (Table 5, Fig. 5) [3 ÷ 7, 21, 22].
Table 5. ICW – Accepting the Call on IP

<table>
<thead>
<tr>
<th>Pre-conditions</th>
<th>Action 1, 2, 3</th>
<th>Action 4</th>
<th>Action 5</th>
<th>Action 6</th>
<th>Action 7</th>
<th>Action 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>The User A is connected to the Internet, an incoming call has arrived to User A, User A chooses to answer the call over IP.</td>
<td>The user A is connected to Internet, an incoming call has arrived for user A, user A chooses to answer the call over IP.</td>
<td>Since a connection to an IVR was still established, the ICW service logic release this connection.</td>
<td>The SSP disconnect the connection to the IVR.</td>
<td>The ICW service logic retrieves User’s A IP address and instructs the network to route the call to the appropriate VoIP gateway and requests to be notified of the outcome of the call routing.</td>
<td>The SSP establish the PSTN connection to the VoIP gateway.</td>
<td>The ICW service logic controls the VoIP gateway to terminate the call to user’s A IP address. Depending on the VoIP gateway used, different control interfaces are possible e.g. H323, MeGaCo.</td>
</tr>
</tbody>
</table>

Post-conditions The call is established between User A and the User B using a VoIP gateway.

4. Conclusions

To complete and substantiate our approach (e.g., service scenario orchestration and modeling), it is essential to make some statements. Accordingly: (1) the innovations flourish in a receptive service environment, so that service groups are commonly at odds with their parent enterprises/businesses. Service derivatives have resulted in a prospering service economy through entrepreneurship and innovation. Therefore, it is important to recognize that service is a business, and that the principles apply equally well to internal and external service organizations. (2) The services are ubiquitous so practically everyone knows what one is. However, what most people do not think about, is that a service is a process, usually a collection of activities to support that process. The activities are organized into components. A component is an organizational entity for instantiating services. Some components provide more than one service and some services are comprised of more than one component. Collectively, the arrangements of components that make up a service offering constitute its architecture. In service architecture, some components are internal people or units, some components are outsourced, and some components are business partners. (3) So, an effective management of service architecture is needed. The main aspect of service management is the choreography of components in a specific business process – that is, how information or tasks is passed between components...
without explicit direction. Another important aspect of service management is keeping track of the components and their attributes [24, 25, 26].

Thus, a practical scenario of the service execution scheme examined - in order to find main imperative aspects of what and how it can influence the entire service management/control framework while applying different service technologies for different service components.

References

3. R. Minerva, C. Moiso, Will the “Circuits to Packets” Revolution pave the way to the “Protocols to APIs” Revolution, 6th Int. Conf. on Intelligence in Networks, Bordeaux, France, January 17-20 (2000) 11.
6. V. Blavette, G. Canal, U. Herzog, C. A. Licciardi, S. Tuffin, EURESM C P090 contribution to PINT and SPIRITS Interaction between Internet and PSTN to request services from one domain to the other, IETF Internet draft, March (2000).
http://www.iec.org/newsletter/august07_2/#analyst_corner
http://www.Webtorias.com