RC23609 (C0505-004) May 13, 2005
Computer Science

IBM Research Report

Unified Communication System Based on Distributed SMIL Browser for Pervasive Computing Devices

Yu Chen Zhou, Yi Min Gan
IBM Research Division
China Research Laboratory
HaoHai Building, No. 7, 5th Street
ShangDi, Beijing 100085
China

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Unified Communication System Based on Distributed SMIL Brower for Pervasive Computing Devices

Yu Chen Zhou Yi Min Gan
IBM China Software Development Laboratory
{zhouyuc, ganyimin}@cn.ibm.com

Abstract

SMIL is a markup language that enables us to describe web based multimedia contents. This paper proposes a flexible design model of distributed SMIL browser for pervasive computing devices whose resources are limited. We present scalable distributed architecture for devices with PvC SMIL engine to play fully featured SMIL based multimedia presentation by leveraging capability of server and intermediate node. According to relations among elements of SMIL document, a method is introduced to decompose SMIL document to a group of internal SMIL models to be distributed to and interpreted on nodes of such system. PvC devices without SMIL interpretation capability could also involve. Based on such distributed PvC SMIL browser, we implement a web based unified communication framework to make the whole communication system a virtual SMIL browser, in which SMIL document is used to define the call logic, and the unified communication and collaboration is implemented as process of web browsing. Comparing with conventional unified communication system, it makes the web technologies including SMIL and web services the core mechanism, and provide more flexibility for integration with web-based applications.

1. Introduction

Synchronized Multimedia Integration Language (SMIL)[1], which is defined by W3C for encoding time based multimedia presentations delivered over the Web, is a widely used standard in different areas such as Web TV, online courses and multimedia presentations. It allows developers to mix many types of media including animation, audio, video, still image, static text and text stream, synchronize them in a timeline, and support user interaction during presentation.

Conventional SMIL browsers such as Xsmiles, Apple QuickTime and Helix by RealNetworks that support full profile and extension of SMIL 2.0, is similar to web browser, which are pure client software designed for PC-like multimedia devices. With advent of more and more networked pervasive computing (PvC) devices, e.g., PDA, mobile phone, in-home and telematics devices, the demand of playing SMIL based multimedia content arises. However, it is hardly to support the presentation of a plurality of media on a single PvC device with limited resources because of following reasons: (1) the media capabilities of such device are only subset of media types defined in SMIL e.g., POTS phone could only support audio, most of PDAs could only support text and still image, and TV and digital HiFi systems could only support real-time video and audio stream (2) The limitation of resource makes it difficult to implement multi-threading, and complex timing and synchronization mechanism.

Great efforts have been made by the industry and academy to enable SMIL browsing on such PvC devices. 3GPP defined SMIL profile for Multimedia Messaging Service [2] that is subset of SMIL 2.0 basic profile. [3], [4] [8], [9] and [10] introduce methods to implement SMIL component for mobile terminals with limited resources. Nevertheless, such methods could not solve the problem of resource and capability completely, and could not support SMIL playing on PvC device without capability of SMIL interpretation.

In this paper, we propose a flexible distributed architecture of PvC SMIL browser to play fully featured SMIL based multimedia presentation by combining a set of PvC devices to provide enough capabilities for SMIL playing. SMIL document is decomposed to a group of internal SMIL models distributed to different nodes of the system for interpretation. It could enable multi-modal interaction through different media channels provided by the PvC devices set. This method has following advantages: (1) on demand composition and dynamic configuration of devices involved in the presentation (2) delivery of synchronized media to a set of devices for some complex scenarios such as home atmosphere control and advanced mobile applications (3) distributed architecture leveraging capability of server and intermediate node instead of conventional pure client SMIL mode (4) involvement of PvC devices without capability of SMIL interpretation.

Based on such distributed SMIL browser, we designed a web based framework that make the unified communication system a virtual SMIL browser in which all the devices involved in a call session build a virtual SMIL browser, and SMIL is used for call control.

The remainder of the paper is structured as follows: section 2 presents the structure of distributed SMIL browser, and the method to decompose and deploy the SMIL model on the nodes of such system; section 3 introduces the principle and architecture of SMIL based unified communication framework; section 4 describes the future work; section 5 provides the main conclusion of this paper.

2. Distributed SMIL Browser for Pervasive Computing Devices

The main goal to design distributed PvC SMIL engine is to make the SMIL document to be presented in distributed system that consists of a set of PvC devices organized with tree-link structure. There are following two kinds of nodes in the system: (1) PvC SMIL engine that is responsible for parsing and interpreting SMIL model and communicate with remote media proxy (proxies) and other PvC SMIL engine(s) (2) remote media proxy that resides on intermediate node or remote media device without SMIL engine, which receives commands from parent PvC SMIL engine to start or stop media rendering, send back events, and provide basic interaction function if necessary. Section 2.1 introduces the method to decompose SMIL document in section; section 2.2 presents the structure of PvC SMIL engine; section 2.3 describe the process to player SMIL document in such system.

2.1 Decomposition and Deployment of SMIL Document

The first step to play SMIL document on distributed SMIL browser is to decompose the SMIL document into parts that could be interpreted independently. In most SMIL presentations, timing elements dominate the hierarchical composition of the document body [6]. SMIL's time containers, <seq>,<par> and <excl>, appear often and are always parents of other elements — typically other timing composites or media object elements. They specify different temporal relationships among their children. The decomposition of the SMIL document mainly focuses on above elements, and other elements are rebound to and deployed with corresponding timing container.

In this system, there are following two categories of SMIL models (1) internal SMIL model to be executed on PvC SMIL engine (2) intermediate model that is transported and deployed on different nodes of the system. In this section we mainly focus on the internal SMIL model that is defined as follows:

$$
\begin{aligned}
&S = \{TC, MO, O, P, B, TR\} \text{ where}\\
&\quad TC = \{tc_1, tc_1, \ldots, tc_m\},\\
&\quad MO = \{mo_1, mo_2, \ldots, mo_n\},\\
&\quad OE = \{oe_1, oe_2, \ldots, oe_p\},\\
&\quad P \subseteq TC \times (TC \cup MO),\\
&\quad R \subseteq (TC \cup MO) \times OE,\\
&\quad TR \subseteq (TC \cup MO) \times (TC \cup MO)
\end{aligned}
$$

TC, MO and OE represent a set of time containers, a set of media objects, and a set of other objects in the SMIL document. P is the parent relation between TC and $TC \cup MO$, e.g., $<tc_l, mo_k> \in P$ means tc_l is parent of mo_k. B is the reference relation between $TC \cup MO$ and O, e.g., $<mo_v, a> \in R$ means video media object mo_v uses region a defined in layout segment. TR is the timing and event relation among $TC \cup MO$, e.g., $<tc_l, tc_k> \in TR$ means tc_l and tc_k has timing or event relation. Media capabilities required by media object mo_i is defined as $CM(mo_i)$, and media capabilities required by time container tc_j is defined as $CM(tc_j) = \cup CM(tc_k)$, where$<tc_j\ tc_k,> \in P$.

The distributed SMIL browser is defined as follows:

$B = \{ N, NR \}$ *where*

$N = \{ n_1, n_2, \ldots, n_q \}$,

$PN \subseteq N \times N$

N presents a set of nodes in the system. PN is the direct parent relation among nodes, e.g., $<n_i, n_j> \in PR$ means n_i is parent of n_j. From perspective of media capability, a node in the system could be defined by a set {CM, CS}, where CM is the set of media capabilities of the node, and CS is capability of SMIL interpretation. For node without capability of SMIL interpretation, CS=False, and $CM=\{t_1, t_2, \ldots t_n\}$ is a set of media types that could be rendered on this node. For any intermediate node n_i, the composite media capabilities $CM_i = \cup CM_k$, where $<n_i, n_k> \in PN_i$.

Media capabilities and SMIL interpretation capabilities are the basis of decomposition and deployment of SMIL document. According t CM and CS of a node in the system, a media object or a time container should be bound and deploy to it. A time container tc_i could be bound and distributed to a child node n_j, only if the CS_j=TRUE and $CM(tc_i) \subseteq CM_j$. A media object mo_k could be bound and distributed to node n_l only if CS_l = FALSE and $CM(mo_k) \subseteq CM_l$. Accordingly, a SMIL document S could be presented by B, only if $CM(S) \subseteq CMB$.

The timing and synchronization relation among decomposed SMIL models must be consistent with that in the original SMIL document. In the normal course of processing, a SMIL 2.0 document's activation hierarchy determines the rendering of document elements. The user can influence the elements selected by using SMIL 2.0 events. The event architecture lets document components that are waiting be activated or terminated to actually start or stop. SMIL 2.0 allows several uses of events, but the most important new semantic in the language is the combination of events and "begin" and "end" attributes. In further combination with the excl element, events provide a powerful mechanism for conditional content activation. Basically, the timing events among time containers and media objects should be kept. So for each time container and media object to be distributed to other node, a local proxy object is generated for linking the two SMIL models.

Also, other SMIL element should be distributed to different nodes according to decomposition of SMIL model. Generally, for any element $oe_i \in OE$, if $<tc_i, oe_i> \in R$, then oe_i should be contained in SMIL model including tc_i. The elements contained in the time container, e.g., <a> element, should be distributed together with the time container. For elements that are not contained in the time container, adjustment is needed to keep the consistency of system behavior and presentation before and after the decomposition. The typical element is <region> defined in layout segment, which is defined for display area for rendering of different media object. According to the rules of decomposition, the <region> elements should be bound and distributed to different internal SMIL models together with the time container that has reference to it. During this process, the value of attributes of the <region> element, e.g., top and left, should be adjusted according to the new display.

The decomposition operation is defined as follows:

$D: S \rightarrow \{S^{'}, SR\}$ *where*

$S' = \{ S'_1, S'_2, \ldots, S'_n \}$,

$SR \subseteq S \times S$

S' is the set of the SMIL models decomposed from S. SR is the parent relation among the decomposed SMIL models.

A simplified sample is shown in figure 1 and 2. The original SMIL document shown in figure 1 contains a video media object, an audio object, a static image for introduction, and a control link to stop the presentation. Figure 2 presents a system to which the SMIL document will be deployed. The root node is a set-top box with digital HiFi system and TV attached, and with a PDA connected via WLAN. There are PvC SMIL engine installed on both the set-top box and the PDA. According to the capabilities of such nodes and predefined binding rule, the original SMIL document is decomposed into two SMIL models: (1) the root model with video and audio media object to be played on TV and HiFi (2) leaf model to be deployed to PDA for introduction and control. To link the two SMIL models, a proxy object is generated in root model. The intermediate model distributed from set-top box is shown in figure 2.

2.2 PvC SMIL Engine

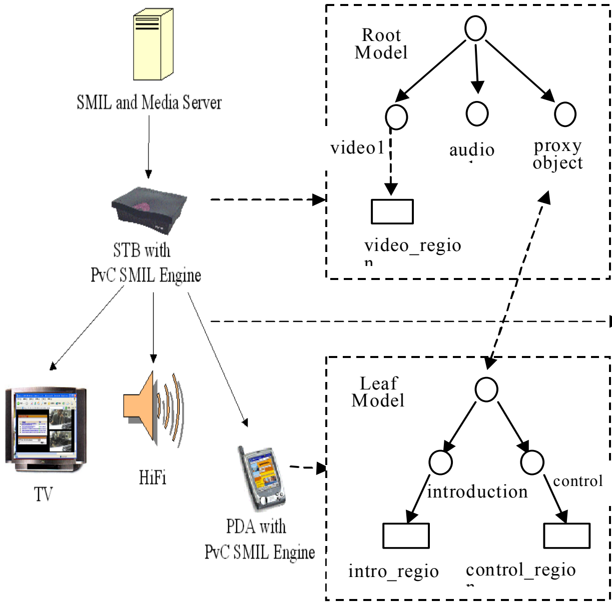
Figure 3 presents the structure of PvC SMIL engine that consists of following components:

Media Device Registry: A PvC SMIL engine could control following types of media components: (1) Local media device that renders media defined by media object in local internal SMIL model (2) Remote media proxy that could be controlled via network to render media remotely. (3) Remote PvC SMIL engine that could be controlled to interpret other internal SMIL model. Each component supports a subset of media capabilities defined by SMIL. The capability and location information of such components are registered in Media Device Registry manually or automatically.

PvC SMIL Parser: While SMIL document is requested to play, this component retrieves SMIL document from server, or intermediate SMIL model from parent PvC SMIL engine. Then it analyzes decompose the SMIL document or model to generate internal SMIL model to be deployed on local PvC SMIL engine, and distribute

```
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <head>
    <layout >
     <region regionName=”video_region” top=”5” left=”5” hight=”480” weight=”640”/>
     <region regionName=”intro_egion” top=” 485” left=” 5” hight=”50” weight=”500”/>
     <region regionName=”control_region” top=” 485” left=”505” hight=”30” weight=”140”/>
    </layout>
  </head>
  <body>
    <par>
      <video id="video1” src=” www.ibm.com/video1.mpg" region=”video_region” begin=”1s”/>
      <audio id="audio1” src=” www.ibm.com/video1.wav" begin="video1.begin"
end=”video1.end”/>
      <par>
        <img id=”introd_img” src=”www.im.net/introduction.gif”/>
        <a href="http:/www.im.net/clip2.smil” alt=”Stop” >
         <img src=”www.im.net/stop.jpg” region=”control_region”>
        </a>
   <par/>
    </par>
  </body>
```

Figure 1. Sample SMIL document




```
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <head>
    <layout >
      <region regionName="intro_region" top=" 5" left=" 5" hight="50"
      weight="500"/>
      <region regionName="control_region" top=" 5" left="505" hight="30"
      weight="140"/>
    </layout>
  </head>
  <body>
    <par>
        <img id="introduction" src="www.ibm.net/intro.gif"/>
        <a href="http:/www.im.net/clip2.smil" alt="Stop" >
          <img id="control" src="www.ibm.net/stop.jpg"
      region="control_region">
        </a>
    </par>
  </body>
</smil>
```

Figure 2. Decomposition of SMIL

intermediate SMIL models children PvC SMIL engine.

SMIL Interpreter: In execution phase, it maintains local timers and control local media device, remote media proxy and leaf SMIL engine to render media and control interaction.

Local Media Device: Some devices with SMIL engine may have local media device. So some of media objects in local internal SMIL model could be rendered locally.

Remote Event Proxy: This component maintains the mapping between proxy objects in local internal SMIL model and media object and time containers distributed to remote SMIL engines and Remote Media Proxy, and is responsible for transportation of serialized events, including SMIL timing events, DOM events [5], and other events between local and remote SMIL engine or Remote Media Proxy.

2.3 Playing SMIL Document

The process to play SMIL document in distributed SMIL browser consists of following two phases: (1) deploying SMIL model recursively (2) interpreting decomposed SMIL models.

Before execution, a capability table should be setup for each PvC SMIL engine involved in rendering to register media and interaction capabilities of children nodes.

When user request to play SMIL document through any interaction channel, the SMIL parser analyzes the SMIL document and generates internal SMIL models. During this process, SMIL parser check the capability information of media components registered in Media Device Registry

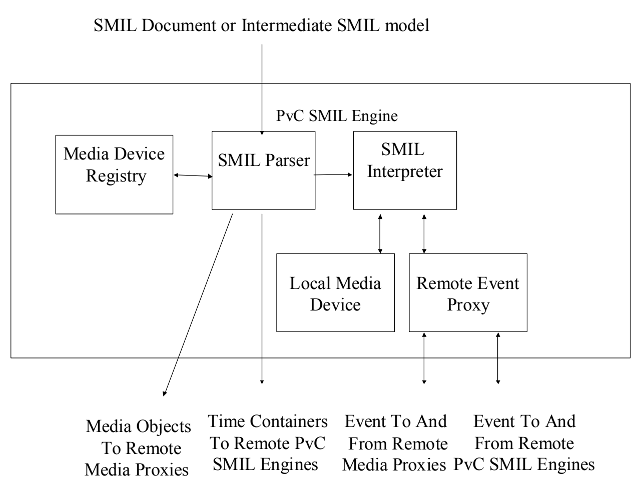


Figure 3. Structure of PvC SMIL engine

and generates each models to be distributed to remote media proxies and PvC SMIL engine according to binding rule. Then SMIL parser will distributed the models to remote devices. The mapping table in event proxy is updated. Finally the SMIL model will be deployed to SMIL interpreter. This process will be executed in the system recursively.

After the SMIL model has been deployed within the whole system successfully. The root PvC SMIL engine starts main timer to invoke all the connected components by sending events to remote device via event proxies. If a proxy object is started, it notifies the corresponding media proxy to play related media or trigger the distributed time container to be started. The whole system interprets SMIL document recursively. During this process, the timing events, DOM events, events for user interaction and other internal events with be passed up and down through event proxy. When main timer stops or any stop criteria is satisfied, the whole process is finished.

3. SMIL Based Unified Communication Framework

Web centered application model plays main role in network application today. Mechanism is needed to integrate variety of communication channels, applications, data types and data sources into a web application framework. Conventional telecommunication systems such as circuit and package switched systems that leverages signaling protocols (e.g., SS7) and specific architecture, is quite different from web-based architecture in the way of controlling the communication session. It is difficult to implement unified communication function and integrate web into its core service. Though the advent of IP based multimedia communication system such as IP Multimedia Subsystem (IMS) [7] defined by the 3GPP, ETSI and Parlay Forum, makes it easy to integrate IP based services into telecommunication system, such systems just extend the functionality leveraging IP technologies, and still follow the architecture and core concepts of conventional telecommunication system such as signaling

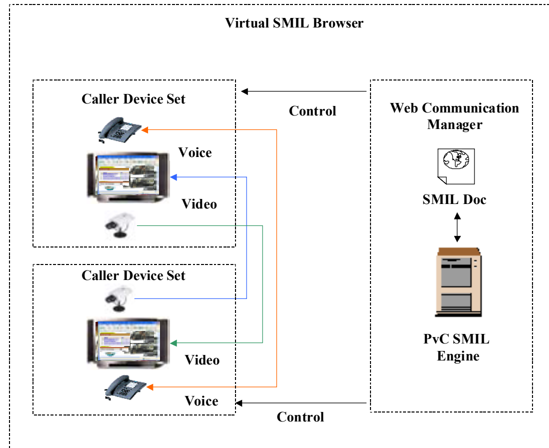


Figure 4. Basic concepts of SMIL based communication

protocol. In such system, web technologies are used on the edge rather than in the core of the system. For example, Web technologies are always used for enhancement of GUI for network management and system configuration.

For SMIL's feature to describe temporal behavior and presentation layout for multimedia objects, it meets the need of following main trends of unified communication system: (1) diversity of communication channels including synchronous channel such as two party call and conferencing, half synchronous channels such as instance messaging and push-to -talk, and asynchronous channels such as email and multimedia message (2) diversity of data type and data source such as video, audio, text based chatting, data transportation, and web based applications (3) more complex collaboration applications such as white board involved in communication system. In this framework, conventional telecommunication systems and other applications are plugged into the system as low layer communication channel supporting specific type of media or data. All the nodes in the system are implemented as web services that are signaling protocol independent.

Leveraging the distributed PvC SMIL engine, we designed a web based unified communication framework to meet the need of applying web technology in communication area. From user perspective, a call in such system means browsing a web based presentation. From system perspective, a call process is playing a SMIL based presentation on a set of multimedia devices including caller device set, callee device set and other device set, such as message server and conference server. Comparing with conventional communication system, it uses SMIL to describe the call logic including synchronization of different communication channel, timing of communication session and process of user interaction, etc. The control process of each call session is implemented by a sequence of invocation of web services. The multimedia message is record of complex scenario including timed media rendering from all communication channels, internal events, user interaction and invocation of applications. The data sources for each terminal are other terminals involved in the communication session. The basic concept of such system is shown in figure 4.

3.1 SMIL Based Call Control

The SMIL document for call control consists parts for all the parties involved in the call session. For example, for a two parties call with multimedia message service, there are 3 scenarios: (1) call and recording message (2) recoding greeting information (3) playing multimedia message.

The typical one is simplified SMIL document for scenario (1) that shown in figure 5. It is for callee's communication manager and contains following contents:

Presentation to caller – It consists of a timer for ringing with period of 15s and two exclusive time containers representing media from callee and greeting message from message server. . The timer is executed firstly, which triggers one of the two

```
<smil xmlns="http://www.w3.org/2001/SMIL20/Language">
  <head>
    <layout type="text/smil-basic-layout">
      <region id="caller_audio_region" width=”640” height=”480”/>
      <region id="callee_video_region” width=”240” height=”240”/>
    </layout>
  </head>
  <body>! 1
    <par>! 2
      <par id="callee-presentation">! 3, to caller
        <audio id="timer" begin= "0s" dur="15s"/>! 5, ringing
        <excl>! 6
          <par id="callee_call" begin="timer.endEvent">! 7
            <audio id="callee_voice" src=”callee-audio_addr”/>! 11
            <video id="callee_video" src=”callee_video_addr” region=”callee_video_region”/>! 12
          </par>
          <par id="message_greeting" begin="timer.end" />! 8
            <audio id=“greeting_voice” src=”msg-audio-addr”/>! 13
           <video id="greeting_video" src= ”msg-video-addr “region=”callee_video_region/>! 14
          </par>
        </excl>
      </par>
      <excl id="caller-presentation">! 4, to callee
        <par id="caller_call" begin="callee_call.beginEvent">! 9
          <audio id="caller-voice" src=”caller-audio_addr” />! 15
         <video id="caller-video" src=”caller_video_addr” region=”caller_video_region” />! 16
        </par>
        <par id="caller_message" begin="message_greeting.endEvent">! 10
          <audio id="caller_voice" src=”caller-audio_addr” / >! 17
         <video id="caller_video" src=”caller_video_addr” region=”caller_video_region” />! 18
        </par>
      </excl>
    </par>
  </body>
</smil>
```

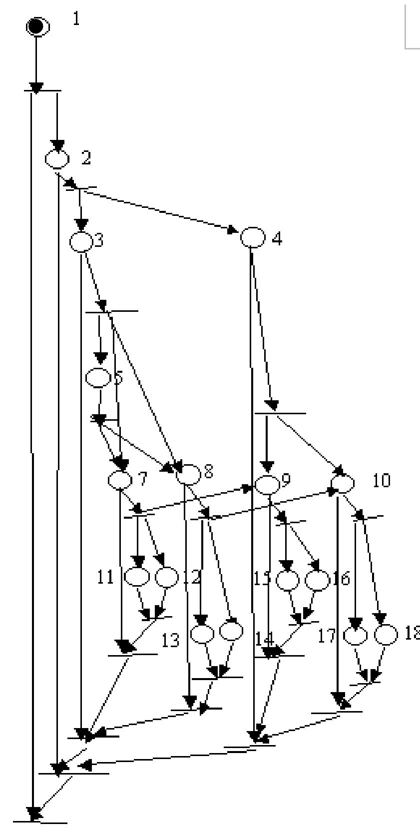


Figure 5 Simplified SMIL document for 2 parties call control

exclusive time container according to user interaction. Once the call is accepted by callee terminal within the ringing period, the "callee-call" timer container is started to control callee terminal to send video and audio data to caller terminal, or the "message-greeting" time container is started to the control the message server to send video and audio data to caller terminal.

Presentation to callee – It consists two exclusive time container representing media from caller to callee terminal or message server. Once the "callee-call" time container is started, the "caller call" time container is started to control caller terminal to send video and audio data to callee terminal. Once "message-greeting" time container is started, the "caller-message" time container is started to control message server to receive and recode video and audio data, and also the timing and synchronization information.

3.2 Architecture

Different from conventional telecom system with 2 layers, signaling and media transmission, the SMIL based unified communication framework has the 3rd layer, web services based control layer

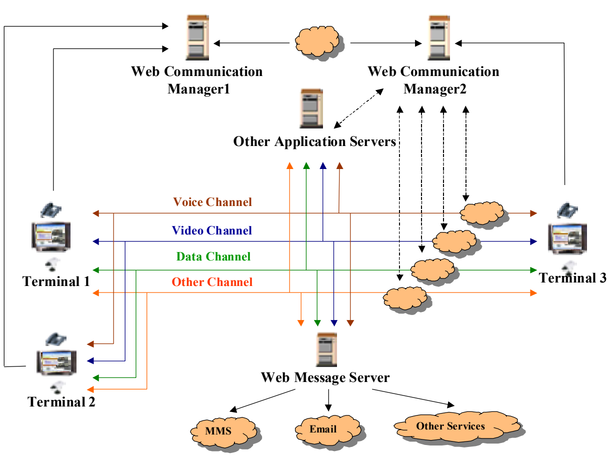


Figure 6. Architecture of SMIL based unified communication system

for compose of the virtual SMIL browser. The conventional communication system, which consists of signaling and media connection, could be taken as one of the channels involved in one communication session. Figure 6 presents the architecture of SMIL based unified communication system. It consists of following components:

Terminal: Component to initiate, response and terminate one or more communication sessions. The terminal could be a device set on behalf of end user or application server control by communication manager to involved in the communication or collaboration process.

Communication Channel: A communication session among two or more parties consists of one or more communication channels. Each channel is one application or communication modal with specific communication protocol and media format, for example, Voice over IP, Instance Messaging, whiteboard based collaboration, or connection to an application server. There are two or more terminals involved in a communication channel, e.g., caller and callee terminals in two parties call, multiple caller terminals in a conference, and client and server of a collaboration application.

Web Communication Manager: Abstractly, it is a backend PvC SMIL engine with additional functions. Logically, it is the container of web based communication logic, and runtime environment for communication session. It has following functions: (1) registers all the devices used as a terminal for end user and control the terminal in communication session (2) interacts with other communication manager to initialize and control one session across multiple communication domains (3) generates and interprets SMIL document for call logic. There should be at least one communication manager within one communication domain.

Terminal for end user consists of one or more devices, and each device provides media and interaction capabilities for one or more communication channels. Each device is mapped to one or more media objects in SMIL document of call logic, and is part of the virtual SMIL browser for the call session. The terminals for end user should be configured manually or automatically before the communication session is initialized. The typical terminal configuration includes those shown in table 1. Other profile could be configured according to real application scenarios, e.g., a user of mobile phone could set up a virtual terminal together with the nearest network projector using UDDI or uPnP in a conference room and then initialize the video call with both video and voice communication channels.

Configuration	Devices
Home Profile	SIP Phone/POTS Phone, STB/SG, TV and Web Camera
Office Profile	IP Phone, IP-PBX, PC with Camera
Mobile Profile	Smart Phone/PDA with GPRS adaptor and Camera

Table 1. Typical configurations of virtual communication terminal

The typical application server is SMIL based Message Server that could record and play back real scenario of caller or callers as SMIL based multimedia message with timing features. It provides following basic functions: (1) connects to caller terminal(s) to record video, audio, data, IM, event and also time relation among such communication channels when there is no response from callee terminal within ringing period (2) connects to callee terminal(s) to plays back recorded scenario of caller or callers as SMIL based multimedia message with timing features when callee(s) request. The SMIL based multimedia message recoded contains a SMIL document generated through recording process, and also media, data and event objects for playing back. The timing relation among different communication channel is recoded with various granularities. In such case, it is not a must for communication channel to have build-in synchronized mechanism, e.g., that between video and audio channels in MPEG.

The process of a two-party call with multimedia message service is described below. To initialize a session, the caller could send request to local communication manager via one of the device in the device set of the caller terminal. The Web communication manager in the caller's domain composes the SMIL presentation of caller terminal, and sent it to communication manager in callee's domain. Web communication manager in the callee's domain generates SMIL documents for callee's terminal and message server in this domain, then compose the SMIL document to be interpreted on callee's communication manager that is shown in figure 6 with all these 3 SMIL documents and local user policies. The SMIL document for call control is deployed on local PvC SMIL engine for interpretation, and the related media objects are rendered on the device set consists of all the devices of caller terminal, and callee terminal or message server.

When the SMIL document for call control is executed, the Web communication manager controls devices of caller and callee terminals to send and receive media streaming directly, or control other node for specific action, e.g., control message server to record and play back multimedia message, and control a IP-PBX to initialize a third party call to connect both IP phones of caller and callee. It also processes event of timing and user interaction. When event are generated by terminals or PvC SMIL engine for termination of the session, the callee content manager notifies all the devices involved in the communication session to stop sending and receiving media and data.

It is easy to add new functions to the system. To support ad hoc conference, the callee terminal should be replaced with a conference bridge. The other web based application server could also become bridge between this framework and other communication system, for example, web message server could convert multimedia message recoded to MMS and email system. To add a communication channel such as file transportation channel among all the terminals in the system, it is easy to plug corresponding communication channel controller in PvC SMIL engine of Web communication manager and add related component with corresponding capabilities in terminals without change on the system structure.

It has following advantages: (1) it supports on demand configuration of communication virtual terminals leveraging capabilities of server in a communication session (2) it is flexibility to integrate other web base applications and switch data sources (3) it is easy to change call control logic and add communication channel (4) it supports asymmetric communication and dynamic switch of communication channels (5) the framework is independent to signaling protocol and data transportation protocol.

4. Future Work

Similar to other web based systems, balance should made between QoS and flexibility of architecture in this system. Because of separation of communication channels, more efforts need to be made to implement synchronization of fine gained media packages in different real-time communication channels, e.g., that between audio packages and video frames. In the prototype, we implemented session level synchronization. More experiments are needed to determine the granularity of the unit to be synchronized for trade-off between user experience and latency caused by control system.

5. Conclusions

We propose a flexible design model of distributed SMIL browser for pervasive computing devices whose resources are limited. We introduce scalable distributed architecture for devices with PvC SMIL engine to play fully featured SMIL based multimedia presentation by leveraging capability of server and intermediate node, and method to decompose and deploy SMIL document on such system.

Based on this technology, we design SMIL based unified communication framework that makes the whole communication system a web based application. It has following features: (1) all the devices involved in a communication session compose a virtual SMIL browser (2) SMIL document is used to define the call control logical including synchronization of different communication channels, timing and process of user interaction (3) the process of unified communication is implemented as a process of web browsing. It takes the advantages of web technologies to provide flexible architecture for unified communication system.

6. References

[1] W3C. "Synchronized Multimedia Integration Language (SMIL 2.0)". TRirmil2Ol, 2001, http://www.w3.org.

[2] OMA. "Multimedia Messaging Service version 1.1". www.openmobilealliance.org.

[3] Satoshi Hieda, Yoshinori Saida, Hiroshi Chishima, Naoki Sato, Yukikazu Nakamoto. "Design of SMIL Browser Functionality in Mobile Terminals". Proceedings of The Sixth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC'03), 2003.

[4] Robert Steele Marcin Lubonski Yuri Ventsov Elaine Lawrence. "Accessing SMIL-based Dynamically Adaptable Multimedia Presentations from Mobile Devices". Proceedings of The International Conference on Information Technology: Coding and Computing (ITCC'04), 2004.

[5] W3C. "Document Object Model". www.w3.org.

[6] Lloyd Rutledge. "SMIL 2.0: XML for Web Multimedia". IEEE Internet Computing, Vol.5, iss.5, pp.78.84.

[7] Lucent Technologies. "IP Multimedia Subsystem Service Architecture". www.lucent.com.

[8] Stuart Goose, Sreedhar Kodlahalli, William Pechter, Rune Hjelsvold. "Streaming Speech3: A Framework for Generating and Streaming 3D Text-To-Speech and Audio Presentations to Wireless PDAs as Specified Using Extensions to SMIL". WWW 2002, May 7-11, 2002.

[9] Ingo Elsen, et.al. "Streaming Technology in 3G Mobile Communication Systems", IEEE Computer, September 2001.

[10] Dick C.A. Bulterman, Jack Jansen, Kleanthis Kleanthous, Kees Blom and Daniel Benden. "AMBULANT: A Fast, Multi-Platform Open Source SMIL Player". MM'04, October 10-16, 2004.