The Call Processing Language: User Control of Internet Telephony Services

Jonathan Lennox
lennox@cs.columbia.edu

February 28, 2000
Motivation

- Internet telephony increasingly replacing circuit-switched network
- Need advanced telephony services for Internet telephony
  - Services comparable to traditional networks
    * Call forwarding, call blocking, time-of-day routing, ...
  - New or enhanced services
    * Interfaces with other parts of Internet
    * Elaborate, custom services
- Need new ways of creating such services
Call Processing Language: Characteristics

- Creatable and editable both by simple graphical tools and by humans
- Independent of underlying signalling protocol
- Safe to run in servers
  - Automatically verifiable when uploaded to server
  - Resource usage (memory, processing time) inherently limited
  - Not Turing-complete: no loops, variables, recursion
  - Can’t interact with inappropriate parts of the server
  - Predictable behavior
- Separate scripts for inbound, outbound, policy
Sample CPL: Graphical Representation

- Acyclic graph: start at root, progress downward
- Follow one output of each node, based on outcome
Representation: XML

- Why XML?
  - Easy to parse
  - Easy to edit
  - Naturally encodes tree structure
    * add simple links for acyclic graph

- Representation specifics
  - Top level tag is call
  - Nodes and outputs are both tags
  - Node parameters are tag attributes
  - link tag connects branches
Example CPL Script

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">

call>
  address-switch field="origin" subfield="host">
    address subaddress-of="example.com">
      location url="sip:jones@example.com">
        proxy>
          busy> <link ref="voicemail" /> </busy>
          noanswer> <link ref="voicemail" /> </noanswer>
          failure> <link ref="voicemail" /> </failure>
        </proxy>
      </location>
    </address>
  otherwise>
    location url="sip:jones@voicemail.example.com"
      id="voicemail">
      redirect />
    </location>
  </otherwise>
</address-switch>
</call>
```
CPL Nodes

Switches
- Address Switch
- String Switch
- Priority Switch
- Time Switch

Signalling Actions
- Proxy
- Redirect
- Response

Locations
- Basic Location
- Location Lookup
- Location Filter

Other Actions
- Mail
- Log

Links
Location Model

- Scripts have an implicit global variable: the location set
- Location nodes add to, replace, or filter location set
- The behavior of **Proxy** and **Redirect** actions is based on the current set of locations
- Allows flexible location handling
(Possible) Downsides to XML

- Location scheme is implicit, not explicit
  - Not possible to have other variables
  - But is this bad or good?

- Some syntax has to be done two-level, moderately ugly
  - e.g. time-switches:
    ```xml
    <time
timeofday="1310-1425,1440-1555,1610-1725"
day="2,4">
    ```

-Verbose
Status

- **Standardization**
  - Work item of IETF IPTel working group, for Proposed Standard
  - Some interest from ITU for H.323 bindings

- **Implementation**
  - Implemented in Lucent “GosSIP” Server
    (but status of this is uncertain)
  - Interest from other Lucent groups
  - In progress for Columbia SIP server
  - Interest from other organizations
Example Scripts
Example: Call Redirect Unconditional

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">
<call>
  <location url="sip:smith@phone.example.com">
    <redirect />
  </location>
</call>
```
Example: Call Forward Busy/No Answer

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">
<call>
  <location url="sip:jones@jonespc.example.com">
    <proxy timeout="8s">
      <busy>
        <location url="sip:jones@voicemail.example.com" merge="clear" id="voicemail" />
      </busy>
    </proxy>
  </location>
</call>
```
Example: Call Screening

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">
<call>
  <address-switch field="origin" subfield="user">
    <address is="anonymous">
      <response status="reject">
        reason="I don’t accept anonymous calls" />
      </response>
    </address>
  </address-switch>
</call>
```
Example: Time-of-day Routing

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">
<call>
  <time-switch>
    <time day="1-5" timeofday="0900-1700">
      <lookup source="registration">
        <success>
          <proxy />
        </success>
      </lookup>
    </time>
    <otherwise>
      <location url="sip:jones@voicemail.example.com">
        <proxy />
      </location>
    </otherwise>
  </time-switch>
</call>
```
Example: Non-call Actions

```xml
<?xml version="1.0" ?>
<!DOCTYPE call SYSTEM "cpl.dtd">

<call>
  <lookup url="http://www.example.com/cgi-bin/locate.cgi?user=jones"
          timeout="8s">
    <success>
      <proxy />
    </success>
    <failure>
      <mail url="mailto:jones@example.com;subject=lookup%20failed" />
    </failure>
  </lookup>
</call>
```
Example: A Complex Example

<call>
  <location url="sip:jones@phone.example.com">
    <proxy timeout="8s">
      <busy>
        <location url="sip:jones@voicemail.example.com" id="voicemail">
          <redirect />
        </location>
      </busy>
      <noanswer>
        <address-switch field="origin">
          <address contains="boss@example.com">
            <location url="tel:+19175551212">
              <proxy />
            </location>
          </address>
        </address-switch>
      </noanswer>
    </proxy>
  </location>
</call>