SECE: Sense Everything, Control Everything

Omer Boyaci, Victoria Beltran and Henning Schulzrinne
SECE allows non-technical users to create services that combine:
- communication
- calendaring
- location
- devices in the physical world

SECE: *event-driven* system
- uses high-level *event languages*
- to trigger action scripts, written in Tcl

IRT Pizza Talk Nov 2010
Not Just an Aggregation Service

Timelines Web Aggregator

IRT Pizza Talk Nov 2010
### Related Work

<table>
<thead>
<tr>
<th>Systems</th>
<th>User rules</th>
<th>User actions</th>
<th>Communications</th>
<th>Time</th>
<th>Location</th>
<th>Presence</th>
<th>Sensors</th>
<th>Web services</th>
<th>Actuators</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECE</td>
<td>NL-like rules</td>
<td>Tcl scripts</td>
<td>Call, email, IM</td>
<td>✔</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>CPL</td>
<td>XML tree</td>
<td>Fixed XML actions</td>
<td>Call</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>LESS</td>
<td>XML tree</td>
<td>XML actions</td>
<td>Call</td>
<td>✔</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
<td>❌</td>
<td>X10, vcr</td>
</tr>
<tr>
<td>SPL</td>
<td>script</td>
<td>Signaling actions</td>
<td>Call</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>VisuCom</td>
<td>Graphical UI</td>
<td>Signaling actions</td>
<td>Call</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>DiaSpec</td>
<td>Java</td>
<td>Java</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
<td>✔ ✔</td>
</tr>
<tr>
<td>CybreMinder</td>
<td>Form based</td>
<td>Reminder</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
</tr>
<tr>
<td>Task.fm</td>
<td>Time rule</td>
<td>Reminder</td>
<td>❌</td>
<td>✔</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
</tbody>
</table>

IRT Pizza Talk Nov 2010
Events & actions

**Events**
- Presence updates
- Incoming calls
- Email
- Calendar entries
- Sensor inputs
- Location updates

**Actions**
- Control the delivery of email
- Route phone calls
- Update social network status
- Control actuators such as lights
- Reminders (email, voice call, SMS)
- Interact with Internet services

IRT Pizza Talk Nov 2010
every sunset {
   homelights on;
}

every week on WE at 6:00 PM{
   email irt_list “Pizza talk at 6:00 PM today.”;
}

if stock.google > 14 {
   sms me "google stock:"+[stock google];
}
Event Rules: More Examples

Time
- Single
  - on February 16, 2010 at 6:00 PM
- Recurring
  - every day at 12:00 until April

Location
- Tom within 5 miles of me

Context
- if my office.temperature > 80

Communication requests
- incoming call
The big picture
UI (Google Web Toolkit - GWT)
User information registry

- root
  - me
    - office
    - location
    - presence
    - phone
  - Bob
    - presence
    - location
  - Alice
    - presence
    - location
  - temperature
  - lights
  - mobile
  - office
  - home
Every day at 12:00 from 01/01/2010 until 04/01/2010 {
    email employees “lunch time” “Location: 5th floor Dinning Room, Time: 12:30”
}

BEGIN:VCALENDAR
BEGIN:VEVENT
DTSTART;TZID=America/New_York:20100101T120000
RRULE:FREQ=DAILY;BYHOUR=12;
    UNTIL=20100401T120000
END:VEVENT
END:VCALENDAR

IRT Pizza Talk Nov 2010
SECE: Location-based rules

user operator location { body }

bob near "Columbia University"
me near 40.807,-73.963

tom within 5 miles of me
me within 3 miles of "2960 Broadway, New York, 10027"

tom in “Rockefeller center”
Me outside of “Manhattan”

bob moved 1.5 miles

- Place types and user-defined locations:

me near a post office
Anne in a museum
me near my tennis club
**SECE: Request-based rules**

*incoming|outgoing* event *from* user|address *to* address *{ body }*

*missed call* *from* user|address *to* address *{ body }*

*received call* *from* user|address *to* address *{ body }*

- **Event:** *call, im, sms*, *voicemail*, *email* (*only incoming*)

- **incoming call**
  - if { [my activity] == "on-the-phone"} forward sip:bob@example.com

- **outgoing call**
  - if {[outgoing destination] == "18003456789"} modify_call destination 12129397054

- **incoming call from Anne**
  - if {[my location] != "office"} auto_answer audio no_office.au –record

- **incoming im**
  - sms me [incoming from] + " sent an im:“ + [incoming content]
SECE: Social Network Integration

Incoming

<table>
<thead>
<tr>
<th>social_network</th>
<th>message_type</th>
</tr>
</thead>
<tbody>
<tr>
<td>facebook</td>
<td>wallmessage</td>
</tr>
<tr>
<td>twitter</td>
<td>newsmessage</td>
</tr>
<tr>
<td>linkedin</td>
<td>direct</td>
</tr>
</tbody>
</table>

social_network status_update

<table>
<thead>
<tr>
<th>social_network</th>
</tr>
</thead>
<tbody>
<tr>
<td>facebook</td>
</tr>
<tr>
<td>twitter</td>
</tr>
<tr>
<td>linkedin</td>
</tr>
</tbody>
</table>

IRT Pizza Talk Nov 2010
Automated Call Handling

• **Control**: Accept, reject, redirect, forward calls based on variety of SECE signals
• **Integration**: Calendar, address book, PSTN, Google Voice, SMS, location, Text-to-speech, voicemail
• **Simplicity**: Natural, easy to learn scripting language
• **Flexibility**: Input from a variety of SECE components involved in call handling
• **Automation**: Scripts for recurring tasks (setup a conf. call based on calendar)

“On mom's birthday, call mom when I am home and near phone.”
“Setup a conference call, enter password, invite people, ring desk phone.”
“If driving and incoming call, play “user driving” and redirect to voicemail.”
“If desk phone ringing and not in room, send SMS with caller's number.”

IRT Pizza Talk Nov 2010
Handling location updates

- **User**
  - publishes his/her location periodically (e.g., every 5 min) to a presence server or to a location service such as Google Latitude

- **Presence server**
  - notifies changes in location to SECE server

- **Google Latitude (or similar service)**
  - SECE retrieves user’s location periodically

- **SECE server**
  - depending on user’s defined rules, queries LoST server

- **LoST server**
  - replies with current information on user’s surroundings

- **SECE server**
  - Takes action based on rules and contextual location information
Name: Turkiye
Type: Popular
Floor: 12
Apt.# in: n
My home country :)

Public or Private

Save Clear Delete Polygon

Center:
(39.02622675, 34.62890625)
Address:
Hacibektas, Nevsehir, Turkey

Location: Bursa, Turkey

Geocoder reply: OK

Found 1 result(s):
1. Address: Bursa, Turkey
OAuth

Diagram:

1. Consumer requests a Request Token (A).
2. Service Provider grants the Request Token (A).
3. Consumer directs the user to the Service Provider (B).
4. User authorizes the request (C).
5. Service Provider directs the user back to the Consumer (C).
6. Consumer requests an Access Token (D).
7. Service Provider grants the Access Token (E).
8. Consumer requests protected resources (G).

Flow:

- A. Request Token
- B. Direct User to Service Provider
- C. Obtain User Authorization
- D. Request Access Token
- E. Grant Access Token
- F. Exchange Request for Access Token
- G. Access Protected Resources
Application sends request to https://api.twitter.com/oauth/authenticate

Is the user logged into twitter.com?

Yes

Is the user authorized the application already?

No

Twitter.com redirects the user back to the application callback with the oauth_token and oauth_verifier

Application completes the OAuth process

No

Sign In Clicked

Twitter.com prompts the user to allow or deny the application access

Deny Clicked

Twitter.com confirms the user cancelled the sign in or denied access

Cancel Clicked

Twitter.com prompts the user to login

No

Yes

Sign In Clicked

Twitter.com prompts the user to allow or deny the application access

Deny Clicked

Twitter.com confirms the user cancelled the sign in or denied access

Cancel Clicked

Twitter.com prompts the user to login

No

Twitter.com confirms the user cancelled the sign in or denied access

Cancel Clicked

Twitter.com prompts the user to login

No

Twitter.com confirms the user cancelled the sign in or denied access

Cancel Clicked
Sensors: smoke, light, humidity, motion, temperature and RFID readers

Actuators: networked devices and actuators such as lights, cameras, sprinklers, heaters, and air conditioners

if my office.temperature > 80 {
    ac on;
}
if my office.smoke equals true {
    sprinklers on;
    sms me "fire in the office";
    call_tts fire-department "fire in the "+[get me.office.address];
    electrical-appliances off;
}
if my warehouse.motion equals true {
    sms me "person in the warehouse."
}
Conclusion

Real world
(location & sensors)

Web services
(SNs, calendar, contacts, ..)

Communication
(VoIP, SMS, email)

SECE

IRT Pizza Talk Nov 2010