# A survey on service creation by end users

Xiaotao Wu and Henning Schulzrinne

{xiaotaow,hgs}@cs.columbia.edu

Computer Science Department, Columbia University, New York, U.S.A.

**Abstract**

We conducted a survey on end users' willingness and capability to create their desired communication services. The survey is based on the graphical service creation tool we implemented for the Language for End System Services (LESS). We call the tool *CUTE*, which stands for *Columbia University Telecommunication service Editor*. This report demonstrates our survey result and shows that relatively inexperienced users are willing and capable to create their desired communication services, and *CUTE* fits their needs.

# 1    Introduction

We defined a service creation language called the Language for End System Services [1] specifically for service creation on telecommunication end systems, such as SIP phones or softphones. One of the design goals of the language is to allow relatively inexperienced users, such as college students, to create communication services by using a graphical or a web-based service creation tool. We have built a graphical service creation tool called *CUTE*, which stands for *Columbia University Telecommunication service Editor*. To evaluate whether we have achieved the design goal of LESS, we conducted a survey on end user service creation.

Survey participants needed to take three steps to complete the survey: first, watch a short movie (2 minutes and 37 seconds) showing an example of using CUTE to create services; second, use CUTE to create services for three scenarios; and third, fill an online form for their background information and service creation experience.

The three scenarios (*scenario 1, 2, and 3*) in the second step have different complexities but all treat incoming calls. *scenario 1* differentiates calls based only on caller's addresses; *scenario 2* differentiates calls based on both caller's addresses and time of calls; and *scenario 3* differentiates calls based on caller's addresses, status of callees, and priority of calls.

The survey participants are divided into three groups: *Group 1* consists of experienced computer users with some knowledge of telecommunication services. This group includes nine people (master or Ph.D. students) from the Internet Real-Time Laboratory (IRT) at Columbia University and one master student from University of Ottwa. They are volunteered to do the survey; *Group 2* consists of experienced computer users that know little about telecommunication services. This group includes five undergraduate

students from the Computer Science Department at Columbia University; and *Group 3* consists of people using computers mainly for text editing and web browsing, including four graduate students from the dental school, business school, medical school, and statistics department respectively, and one faculty member from the biostatistics department. We pay *group 2* and *3* participants ten dollars each for doing the survey. The survey result shows that the survey participants are willing and capable of creating end system services by their own. Below we present and analyze the survey results regarding how many participants would like to create services by their own, and whether they can correctly create services for scenario 1, 2, and 3, what services they are interested, whether they like CUTE, whether they can understand LESS, and whether they are aware of feature conflicts and how they would like to detect and resolve feature conflicts.

Because this survey requires survey participants' background information, such as the department they are working at, we had the survey reviewed by Columbia University Institutional Review Board (IRB). IRB concluded that this survey was not human subjects research thus did not require further review by IRB (IRB AAAA5250 (Y1M00)).

## 2     Users' willingness to create services

Figure 2.1 shows how many participants would like to create their own services. Overall, $70\%$ participants consider that they will create all their desired services. This is an unexpected result to me as presumably users cannot and should not create complicated services. We expected that most participants would create simple services by themselves but ask professionals for complicated ones. However, the result shows that only $15\%$ of the participants would like to ask professionals to create complicated services. One

**Do you like to create your telecommunication services by your own or pay a professional to create them?**

1. I would like to create all my services.
2. I would like to create some of my services, but pay a professional to create some complicated services for me.
3. I would like to pay a professional to create all services because that's more reliable.
4. I do not need to use any supplemental telecommunication services at all. I only need direct end-to-end call.
5. Other

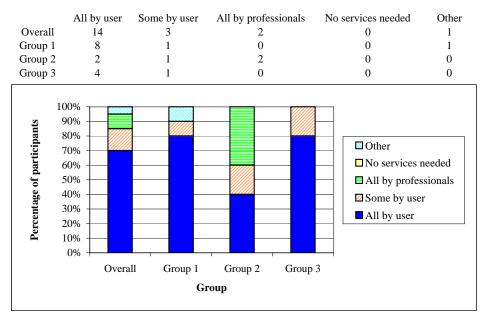|         | All by user | Some by user | All by professionals | No services needed | Other |
|---------|-------------|--------------|----------------------|--------------------|-------|
| Overall | 14          | 3            | 2                    | 0                  | 1     |
| Group 1 | 8           | 1            | 0                    | 0                  | 1     |
| Group 2 | 2           | 1            | 2                    | 0                  | 0     |
| Group 3 | 4           | 1            | 0                    | 0                  | 0     |



Figure 1: Users' willingness to create services

explanation is that most participants do not need complicated services. Another explanation is that they had not thought about any complicated services when doing the survey, but they really need. No matter what the explanation is, the result reveals one fact that most participants (overall $85\%$ of the participants, including $90\%$ *group 1* participants, $60\%$ *group 2* participants, and $100\%$ *group 3* participants) are willing to create all or part of their desired services.

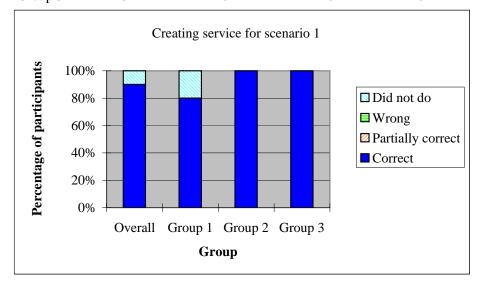|  | Correct | Partially correct | Wrong | Did not do |
| --- | --- | --- | --- | --- |
| Overall | 18 | 0 | 0 | 2 |
| Group 1 | 8 | 0 | 0 | 2 |
| Group 2 | 5 | 0 | 0 | 0 |
| Group 3 | 5 | 0 | 0 | 0 |

Figure 2: Service creation for scenario 1

# 3    Users' capability to create services

Users' willingness itself cannot ensure successful service creation. Users' capability in service creation is also important. The second step of the survey has three scenarios to evaluate users' service creation capability. Below is the description of the three scenarios: *Scenario 1* – "When John Smith calls me, forward the call to 12123334444"; *Scenario 2* – "I have a meeting on 05/30/2006, from 9:00AM to 11:00AM. For an incoming call during the meeting, if the call is from my boss, John Smith, I will answer the call. Otherwise, I will forward the call to my voicemail at 12123334444"; *Scenario 3* – "For an incoming call while my activity is `on the phone`, if the call is not from my boss, John Smith, and is not an emergency call, I will reject the call".

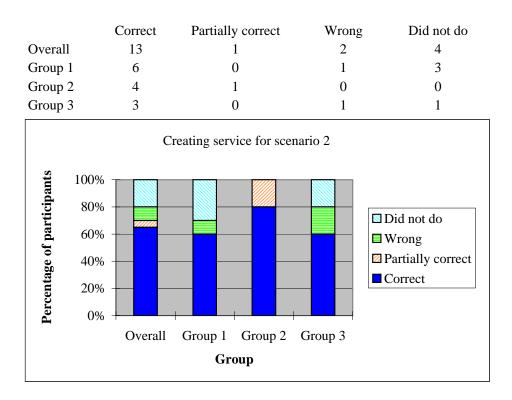| | Correct | Partially correct | Wrong | Did not do |
|---|---|---|---|---|
| Overall | 13 | 1 | 2 | 4 |
| Group 1 | 6 | 0 | 1 | 3 |
| Group 2 | 4 | 1 | 0 | 0 |
| Group 3 | 3 | 0 | 1 | 1 |



Figure 3: Service creation for scenario 2

Figure 3.1 shows that overall 90% of the participants (80% for *group 1*, 100% for *group 2*, and 100% for *group 3*) can correctly create the service for *scenario 1*. If we exclude the participants who did not create the service (10%), all participants can correctly create the service for *scenario 1*. We can then conclude that end users with some computer experiences can handle simple service creation, such as handling calls based only on the caller's address.

Figure 3.2 shows that overall 65% of the participants (60% for *group 1*, 80% for *group 2*, and 60% for *group 3*) can correctly create the service for *scenario 2*. If we exclude the participants who did not create the service, overall 81% of the participants can correctly create the service.
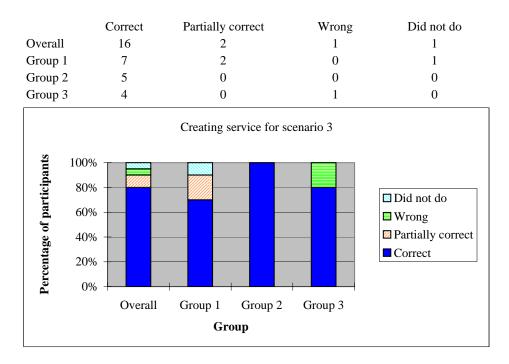
|          | Correct | Partially correct | Wrong | Did not do |
|----------|---------|-------------------|-------|------------|
| Overall  | 16      | 2                 | 1     | 1          |
| Group 1  | 7       | 2                 | 0     | 1          |
| Group 2  | 5       | 0                 | 0     | 0          |
| Group 3  | 4       | 0                 | 1     | 0          |



Figure 4: Service creation for scenario 3

Figure 3.3 shows that overall $80\%$ of the participants ($70\%$ for *group 1*, $100\%$ for *group 2*, and $80\%$ for *group 3*) can correctly create the service for *scenario 3*. If we exclude the participants who did not create the service, overall $84\%$ of the participants can correctly create the service.

The service creation results of *scenario 2* and *3* prove that end uses can create some more complicated services, but not every user can correctly create these services. Hence, it is important to define and design viable ways to find out potential incorrect services, such as by learning from users' call history, or by simulating use cases.

**Do you feel comfortable to use CUTE to create your telecommunication services?**

1. I feel very comfortable to use CUTE to create my services.
2. I can easily use CUTE to create my services, but it requires some improvements to make the tool more useful and friendly.
3. I have some difficulties in using CUTE, but with some additional training, I am confident that I can use the tool.
4. I do not know how to use CUTE, and additional training will not help.
5. Other

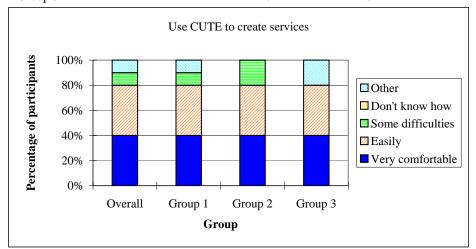| | Very comfortable | Easily | Some difficulties | Don't know how | Other |
|---|---|---|---|---|---|
| Overall | 8 | 8 | 2 | 0 | 2 |
| Group 1 | 4 | 4 | 1 | 0 | 1 |
| Group 2 | 2 | 2 | 1 | 0 | 0 |
| Group 3 | 2 | 2 | 0 | 0 | 1 |



Figure 5: Using CUTE to create services

# 4 Evaluating CUTE

Figure 4.1 shows that overall $80\%$ of the participants ($80\%$ for all groups) can easily use CUTE to create services. On one hand, this shows that CUTE has a relatively easy to use interface. On the other hand, this also proves that end users can create communication services if there is an appropriate service creation tool. In addition, this proves that the tree-like representation of services is acceptable to users because CUTE's user interface presents services in decision trees.

# 5    Evaluating LESS

**Do you think that the "source code" of the services is easy to understand?**

1. Yes, the "source code" is easy to understand.
2. I can only partially understand the "source code".
3. I cannot understand the "source code".
4. I had not read the "source code" of the services at all.
5. Other

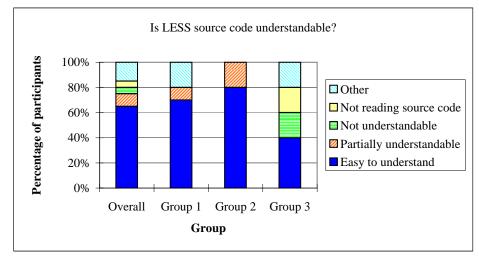|  | Easy to understand | Partially understandable | Not understandable | Not reading source code | Other |
|---|---|---|---|---|---|
| Overall | 13 | 2 | 1 | 1 | 3 |
| Group 1 | 7 | 1 | 0 | 0 | 2 |
| Group 2 | 4 | 1 | 0 | 0 | 0 |
| Group 3 | 2 | 0 | 1 | 1 | 1 |



Figure 6: Understanding LESS source code

Since CUTE is based on LESS, we also ask survey participants to evaluate LESS. In general, end users need not to read service source code. However, because we have noticed that HTML's ease of learning and the view source capability for browsers has bootstrapped the Web's popularity in an amazing way, we think the readability of LESS is very important. Thus, we ask survey participants to read the source code of the services they created, but we did not train them on LESS. Figure 5.1 shows that many survey

participants (overall $65\%$ of the participants, $70\%$ for *group 1*, $80\%$ for *group 2*, and $40\%$ for *group 3*) can easily understand the source code of the services they created, even without any training. It proves the readability of LESS.

# 6    Services of interest

| | Very interested (score: 3) | Somewhat interested (score: 2) | Useless (score: 1) | Not understandable (score: 0) | Average |
|---|---|---|---|---|---|
| location-based | 16 | 4 | 0 | 0 | 2.8 |
| time-based | 13 | 5 | 2 | 0 | 2.55 |
| user-based | 12 | 6 | 2 | 0 | 2.5 |
| email alerts | 11 | 7 | 2 | 0 | 2.45 |
| changing context | 11 | 6 | 3 | 0 | 2.4 |
| priority-based | 11 | 7 | 1 | 1 | 2.4 |
| changing status | 10 | 7 | 2 | 1 | 2.3 |
| presence-based | 10 | 6 | 3 | 1 | 2.25 |
| language-based | 0 | 5 | 13 | 2 | 1.15 |

Table 1: User-interested services

In addition to collecting information on users' service creation experience, the survey also asks the participants on what services they are interested in. Table 6.1 shows the answer. In the table, *location-based* means handling calls based on users' physical locations, e.g., *" Automatically change the ring style to vibrating when I am in a movie theatre"*; *time-based* means handling calls based the time, e.g., *"Forward calls to voice-mail before 8:00AM"*; *user-based* means handling calls based on the caller or the callee's address, e.g., *"Forward calls to voicemail if Alice calls"*; *email alert* means notifying users by email for incoming calls; *changing context* means changing the communication environment for a conversation, e.g., *"Pause Windows media player when accepting an*

*incoming call"*; *priority-based* means handling calls based on the priority of the call, e.g., *"Forward urgent calls to my cell phone, otherwise, to voicemail"*; *changing status* means detecting and updating users' status information automatically, e.g., *"Change my status to `on the phone` when I accept a call"*; *presence-based* means handling calls based on presence status, e.g., *"Reject calls when my presence status is `busy`"*; *language-based* means handling calls based on users' language preference, e.g., *"Reject non-English calls"*. Users are most interested in location-based services, and least interested in language-based services. Note that none of the survey participants were working in an enterprise environment. For enterprise users, the answer may be different. For example, language-based services are not very useful for individual residential users, but can be very useful for a customer service center.

# 7    Handling feature conflicts

Correctly handling feature conflicts is very important for successful service creation. We claim that end users must be involved in feature conflict resolution, which requires end users to understand feature conflicts. Hence, the survey asks survey participants whether they can understand feature conflicts or not. Figure 7.1 shows that overall $50\%$ of the participants ($50\%$ for *group 1*, $60\%$ for *group 2*, and $40\%$ for *group 3*) were aware of the feature conflicts we designed without any hint, and overall $40\%$ participants ($30\%$ for *group 1*, $40\%$ for *group 2*, and $60\%$ for *group 3*) can understand the conflicts with some hint. The result suggests that it is practical to present feature conflicts to users so as to involve users in resolving the conflicts.

There are many different ways to involve users in resolving feature conflicts.

**You may create multiple services to handle different situations. For example, you may create a service to automatically answer calls from your boss. If you have a meeting tomorrow, you may then create another service to automatically reject calls during the meeting. Are you aware that there is a conflict between these two services**?

1. Yes, I noticed the conflict as soon as I saw these two services, and could understand the conflict.
2. Yes, I can understand the conflict after I read the following hint. (Hint: If your boss calls you during the meeting, the first service will answer the call, but the second service will reject the call).
3. No, I cannot understand the conflict.
4. I cannot understand the question at all.
5. Other

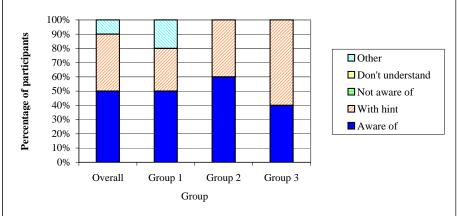| | Aware of | With hint | Not aware of | Don't understand | Other |
|---|---|---|---|---|---|
| Overall | 10 | 8 | 0 | 0 | 2 |
| Group 1 | 5 | 3 | 0 | 0 | 2 |
| Group 2 | 3 | 2 | 0 | 0 | 0 |
| Group 3 | 2 | 3 | 0 | 0 | 0 |



Figure 7: Being aware of feature conflicts

Users may detect feature conflicts by themselves and manually resolve the conflicts. They may also use a tool, such as CUTE, to help them detect and resolve the conflicts. There are two survey questions investigating users' preference on detecting and resolving feature conflicts.

Figure 7.2 shows that overall $85\%$ survey participants ($70\%$ for *group 1*, $100\%$ for *group 2*, and $100\%$ for *group 3*) would like to use CUTE to detect feature conflicts. Figure 7.3 shows that overall $80\%$ of the participants ($90\%$ for *group 1*, $80\%$ for *group 2*,

**For the conflicting services, how would you like to have the conflict be detected?**

1.  By myself.
2.  By the service creation tool, such as CUTE.
3.  By other professional service creators.
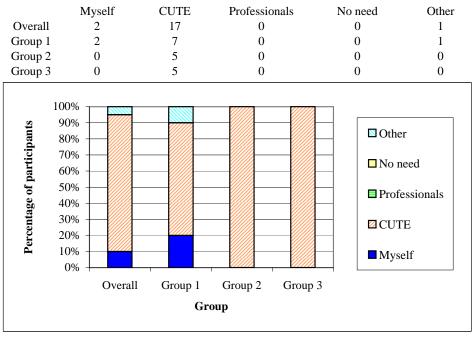4.  No need to detect the conflicts at all.
5.  Other

|         | Myself | CUTE | Professionals | No need | Other |
|---------|--------|------|---------------|---------|-------|
| Overall | 2      | 17   | 0             | 0       | 1     |
| Group 1 | 2      | 7    | 0             | 0       | 1     |
| Group 2 | 0      | 5    | 0             | 0       | 0     |
| Group 3 | 0      | 5    | 0             | 0       | 0     |



Figure 8: Detecting feature conflicts

and $60\%$ for *group 3*) would like to handle feature conflicts based on the choices provided by CUTE, and $15\%$ of the participants would like CUTE to automatically handle feature conflicts. Based on Figure 7.2 and 7.3, it is important to develop a feature interaction handling algorithm in CUTE to handle LESS-based feature conflicts.

**For the services conflicting, how would you handle the conflict?**

1. I will manually change my services (e.g., manually set the second service with a higher priority than the first service so calls during the meeting will be rejected).
2. I would expect CUTE to provide some options (e.g., if your boss calls you during the meeting, option 1, answer the call; option 2, reject the call; option 3, forward the call to my voicemail) to me so I can select one option for handling the conflict.
3. I would expect CUTE to learn from my previous communication behaviors and automatically resolve the conflict for me.
4. I did not see any problems for the conflict so I do not need to handle the conflict.
5. Other

|         | Manually | Provided with choices | Automatically by CUTE | No need | Other |
|---------|----------|-----------------------|-----------------------|---------|-------|
| Overall | 1        | 16                    | 3                     | 0       | 0     |
| Group 1 | 1        | 9                     | 0                     | 0       | 0     |
| Group 2 | 0        | 4                     | 1                     | 0       | 0     |
| Group 3 | 0        | 3                     | 2                     | 0       | 0     |



Figure 9: Handling feature conflicts

# 8    Summary

In summary, the survey shows that relatively inexperienced users are willing and capable to create their desired services, and our LESS-based service creation tool, CUTE, fits their needs. In addition, many users can easily understand LESS source code. The survey also reveals that users can understand feature conflicts and would like to resolve feature conflicts based on the choices provided by service creation tools, such as CUTE.

# Bibliography

[1] Xiaotao Wu and Henning Schulzrinne. LESS: language for end system services in Internet telephony. Internet Draft draft-wu-iptel-less, IETF, February 2005. Expired.