Feature Interaction Problem in Computer-Telephony Integration Systems

Nikolay Anisimov[1], Alec Miloslavsky, Gregory Pogosyants[2]
Genesys Telecommunication Labs, Inc.
1155 Market St., San Francisco, CA 94103, USA
E-mail: {anisimov,alec,gregp}@genesyslab.com
WWW: http://www.genesyslab.com

Abstract. In this paper we describe an ongoing project aimed at investigating the impact of feature interaction problems on computer telephony integration (CTI). As a representative and sophisticated example of CTI we use a call center. We outline a typical structure of a call-center that comprises computing and telephony equipment, software and personnel. We then discuss the specific features of call centers to show how they influence the feature interaction issue in a new framework. We also discuss possible ways to cope with feature interaction problems on three levels: the requirement, specification and implementation levels. In particular we stress the need to flexibly specify a manipulation with telephone features during a call processing and suggest using a graphical icon language to specify call center operator behavior.

1. Introduction

Currently, interest in computer-telephony integration (CTI) is growing rapidly. CTI can be defined as a technology directed at enabling the creation of business applications that actively use areas of telephony and computing [12]. A wide spectrum of CTI applications has been developed, ranging from simple desktop integration (e.g. screen-based telephony) to complex call center applications involving a large amount of telecommunication and computer equipment and hundreds of operators. Our focus is to consider the latter type of CTI applications, which to some extent can be perceived as a general case for other CTI applications.

The typical call center employs agents to process customer calls. This call processing typically involves access to data from computer systems (e.g. databases and e-mail), other devices (e.g. fax and IVR), and communication with other agents. The latter is necessary when the current agent is unable to continue call processing or when an agent must transfer the call to a manager or to another agent better suited to answer the client's questions. The actions of the agent during the call processing are heavily regulated by specific scenarios, specially developed for such type of calls, referred to as 'scripts'. The same agent can work with varying call types, controlled by various scripts.

The extensive experience in call center software development accumulated by Genesys Telecommunication Laboratories, Inc., reveals that some of the difficulties that appeared during the development and maintenance process are results of the feature interaction

[1] Also with the Institute for Automation & Control Processes, Russian Academy of Sciences
[2] Also with *the Institute of Informanics Problem, Russian Academy of Sciences*

problem inherent in the telephone world and therefore inherited by CTI. At the same time it is clear that the some characteristic specific to CTI systems may change the manifestation of the problem of feature interaction.

In this paper we describe an ongoing project aimed at investigating the impact of feature interaction on computer telephony integration, and at developing techniques for solving the problem. At present, the project discussed is at the initial stage. More specifically, in this paper we outline the typical structure of a call center and stress its specific features with respect to IN in light of the feature interaction problem; point out sources of feature interactions in CTI; and discuss some possible ways of solving the problems.

2. Structure of call centers

Figure 1 shows a typical structure of a call center. A call center comprises a telephone switch connected with a telephone network and a computer called telephony server (T-Server for short) with the aid of CTI link.

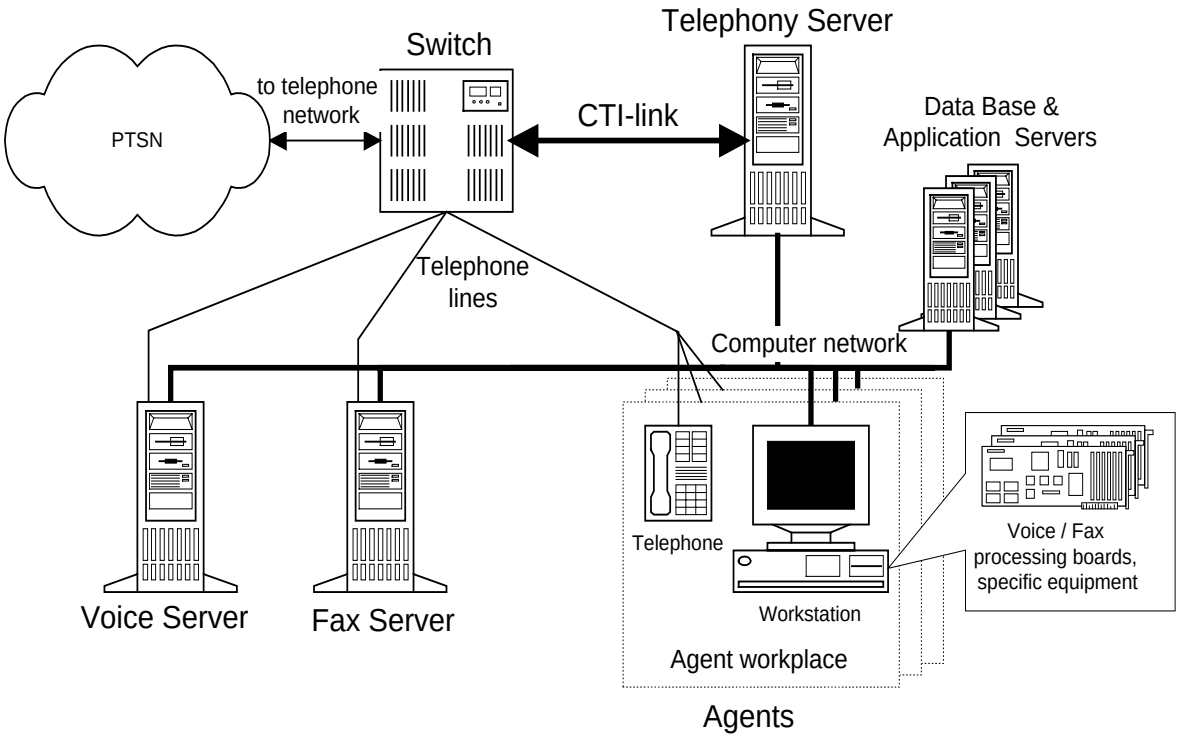

Figure 1: Call center environment

The communication between the switch and T-Server via the CTI link is regulated by the CTI protocol. The CTI protocol defines message interchange so that T-Server can track switch processes while at the same time exercising control over them.

The T-Server is a key element of a call center, since it serves as a bridge between telephony and computing worlds. More specifically, T-Server provides applications with telephone services while communicating with switch via the CTI protocol. The functions of T-Server include:

- Associating user data with a call, to accompany the call through its whole life cycle.
- Providing applications with a simpler and more convenient telephone interface than that offered by the CTI link alone. For instance, T-Server hides a many auxiliary events and messages from applications.

- Detecting and resolving errors and incorrect situations causes when the CTI protocol does not give an adequate picture about processes in the switch. Some of the errors can be qualified as feature interaction problems. For example, sometimes T-Server can not decide to which feature an incoming event belongs, due to insufficient information in the event's parameters. Such errors are usually detected and resolved by developers on an ad hoc basis.

Let us discuss some specific call center characteristics that distinguish it from conventional telephony systems and are relevant to the feature interaction problem.

CTI link and CTI -protocol. At present almost all switches have their own CTI protocol. Different switches may provide different interpretation and depth of call tracking. However the CTI link language and protocol are usually vendor-specific. Using CSTA [4] and SCAI [1] standards for describing switch behavior is a big step in right direction, but still CSTA/SCAI-based CTI link protocols differ from each other because of real difference in behavior of the switches themselves. Moreover, most switches use different event flow on the CTI link, depending on how calls are processed: manually or with CTI control. All this means that we can expect the full spectrum of features interaction and at the same time have a good opportunity to resolve it on CTI server level. CTI has significantly extended and improved call model of call center behavior. Associating the call information coming from the switch with the connection information and business data stored in the Server repository is a key component of this new concept.

Agents and customers. The role of the customer in a call center is played by an operator called an agent. Agents differ from usual customers in conventional telephony. First, their use of a telephone is extensive and strictly regulated by a scenario of call processing, called a script. Depending of the evolution of a dialog, an agent may use one or another feature of the script, such as making a consulting call, call transfer, making a conference call, etc. Second, typically agents do not need the whole spectrum of features designed for IN. For instance, agents do not use a Call Waiting feature because the system itself provides a call transfer to any available agent. In sum, we can say that the agent does not need a wide array of features but does need flexibility in their utilization (e.g. by programming their sequence).

GUI interface. Like multimedia systems [10], the integration of telephony and computers allows a radical reduction in feature interactions caused by limitations of customer premise equipment signaling capabilities [3]. Indeed, CTI provides the ability to move telephony buttons to the computer screen and extend their number and therefore the number of signals.

3. Possible solutions

Similar to [5] we will distinguish three levels on which a struggle with feature interactions can be waged. We will refer to these levels as service, protocol and implementation levels. We also consider possible approaches to tackling the problem of feature interactions during call center software development.

Service level. As we mentioned earlier, an agent needs a flexible way to manipulate telephony features. Moreover, it is highly desirable that manipulations with features could be seamlessly incorporated into a script that regulates agent behavior during call processing. Here the agent-based model reported in [8] can be extremely useful. That approach allows you to build different feature configurations, called *policies*, for each customer. It this case for each type of call that is processed by a group of agents we can build a script containing a policy of manipulations with telephony features. At present for agent scripting we use a graphical icon language with a corresponding script editor and engine developed in Genesys Telecommunication Laboratories, Inc. These tools can be

also be used to program agent policies using a convenient GUI interface. In Figure 2 we present a screen shot of the editor.

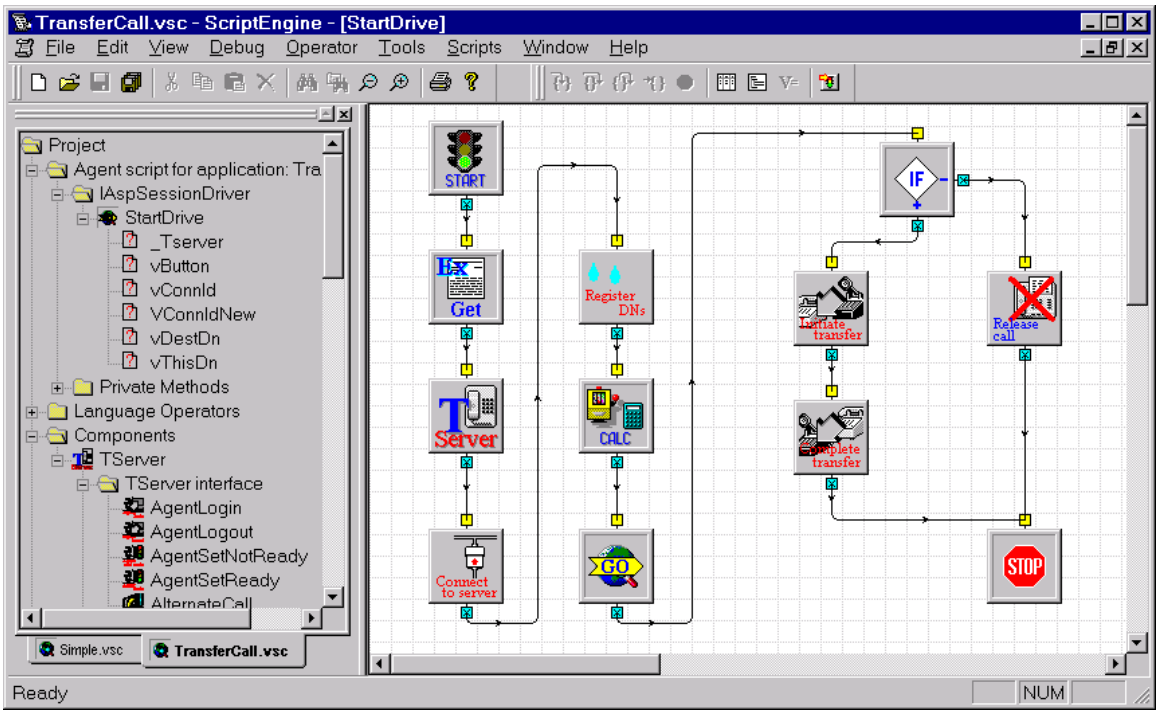

Figure 2: Example of agent script

Note that the agent behavior can be defined not only by a script of a call but also by other scripts (policies), such as general scripts (e.g. department policy), as well as personal policy.

Protocol level. On this level the problem of feature interaction can be formulated as detection and resolution of errors caused by feature interaction and shortcomings of CTI-protocols. Because we can not change the CTI protocols, T-Server must fulfill functions of detection and resolution of feature interaction during run time. Here attention should be paid to online methods of feature interaction detection and resolution. We can point out a behavior based approach to run-time feature interaction detection and resolution [11] that alleviates any need for priory information about features. We can also mention other approaches [2,6] which can be useful in solving our problem. Clearly at this level formal techniques and corresponding tools can provide substantial help. We can point out standard formal description techniques like SDL, MSC, and LOTOS that are extensively used in attacking the problem of feature interaction [2,7,9].

Implementation level. The result of efforts on the protocol level is a (possibly formal) specification of a subsystem in T-Server responsible for feature interaction detection and resolution. The incorporation of such a subsystem into the software may be additional source of errors, although we can reduce this risk by using systematic programming tools and technologies such as object oriented ltechniques (e.g. OMT, UML, SDT) and testing processes [5].

Concluding remarks

In this paper we discussed the problem of feature interaction in the context of computer-telephony integration systems; and we have shown that features of telephony world transfer that problem into the realm of CTI, while its treatment looks somewhat different.

References

[1] ANSI T1.626-1993. Switch-Computer Application Interface (SCAI)

[2] I.Aggoun, P.Combes, Observers in the SCE and the SEE to detect and Resolve Service Interaction. In: P. Dini, R. Boutaba, L.Logrippo. Feature Interaction in Telecommunication Networks IV. IOS Press, 1997, pp.254-270.

[3] E.J. Cameron, N. Griffeth, Y.-J. Linand, M.E. Nilson, W.K. Schnure, and H. Velthuijsen. A Feature Interaction Benchmark for IN and Beyond. In: L. G. Bouma, H.Velthuijsen, (Eds.) Feature Interactions in Telecommunications Systems, Amsterdam, The Netherlands, May 1994. IOS Press, pp. 1-23.

[4] ECMA 179. Services for Computer-Supported Telecommunication Application, 1992.

[5] J. C. Godskesen. A formal framework for feature interaction with emphasis on testing. In K. E. Cheng and T. Ohta, ed. Feature Interactions in Telecommunications III, Tokyo, Japan, Oct 1995. IOS Press., pp. 21-30.

[6] Y. Kakuda, A. Inoue, H. Asada, T. Kikuno, and T. Ohta. A dynamic resolution method for feature interactions and its evolution. . In K. E. Cheng and T. Ohta, ed. Feature Interactions in Telecommunications III, Tokyo, Japan, Oct 1995. IOS Press, pp. 97-114.

[7] B.Kelly, M.Crowner, J.King, Feature Interaction Detection Using SDL Models. IEEE Globecom 1994.

[8] M. Rizzo, I.A. Utting. An Agent-based Model for the Provision of Advanced Telecommunication Services. In: Proc. of the TINA'95 Conference: "Integrating Telecommunications and Distributed Computing -- from Concept to Reality", Melbourne, Australia, 1995, Vol.1, pp. 205-218. IEEE Communication Society.

[9] B Stepien and L Logrippo. Feature interaction detection by using backward reasoning with LOTOS. In S.T. Vuong and S.T. Chanson, editors, Protocol Specification, Testing and Verification XIV, pp. 71-86. Chapman & Hall, 1995.

[10] S Tsang, E H Magill, and B.Kelly. The feature interaction problem in multimedia services: Past, present and future. University of Strathclyde, Apr. 1996. 4th Deliverable.

[11] S. Tsang and E. H. Magill. Bahaviour Based Run-Time Feature Interaction and Resolution Approach For Intelligent Network. In: P. Dini, R. Boutaba, L.Logrippo. Feature Interaction in Telecommunication Networks IV. IOS Press, 1997, pp.254-270.

[12] R.Walters. Computer Telephone Integration (Artech House, 1993)