I.S.F.
Improved Systems Factors

WHY MOST CTI IMPLEMENTATIONS FAIL

BY MALCOLM HAMER

Introduction

Computer/Telephone Integration (CTI) is a term that covers a range of different arrangements, used in Call Centers and Helpdesks, for tying together the operation of some or all of the following components:

- A telephone switching system - typically an Automatic Call Distribution system (ACD), which is either a standard PBX with ACD software modules installed, or a PBX-like switch that is dedicated to ACD operation
- The telephone instruments (telsets) used by the agents
- The agent workstations
- One or more computer systems (mainframes or servers) that hold customer data and process customer transactions
- An Interactive Voice Response (IVR) system – either a basic one that recognizes only touch-tone responses from the caller, or a more sophisticated one that includes speech recognition.

A CTI arrangement ties together the operation of some or all of these components in order to:

- Provide better service to customers who call the call center/helpdesk, and
- Maximize agent productivity in the call center/helpdesk.

The telsets are connected to the ACD via connections that carry voice signals. Also, there are voice-carrying connections between the IVR and the ACD. (These voice connections may be analog, digital, or voice-over-IP: the voice technology makes no difference to the overall architecture of the CTI solution.) In parallel with these voice-carrying connections between the telsets, the ACD, and the IVR, there are data-carrying connections linking together the agent workstations, the computer system, the ACD, and the IVR. These data connections typically take the form of a LAN (provided that they are all at the same site). If one of the components, such as the computer system, is at another site then a private IP wide-area network (WAN) is used to link the remote component to the other components.

IVR terminology

Various names are given to voice response systems by vendors. Even though some vendors may make subtle distinctions between them, the terms IVR (Interactive Voice Response system), AVR (Automatic Voice Response system), and VRU (Voice Response Unit) tend to be used interchangeably.

Another common term is NIVR, which is short for Network Interactive Voice Response service. This is a voice response service provided by a long distance carrier or local telephone service provider, using equipment on that organization's premises. For an NIVR to be part of a CTI arrangement it must be linked to the on-site systems by IP network connectivity. From a telephony point of view, an NIVR is typically connected direct to the carrier's network to receive incoming calls. In other words, the calls go to the NIVR without first passing through the ACD. The carrier provides the switching functions required to pass the call to the ACD *after* the caller has interacted with the NIVR. This arrangement severely limits the flexibility of the CTI arrangement. More will be said about this later.

When does CTI exist?

There is a spectrum of possible telephony arrangements for a call center/helpdesk, ranging from "definitely not CTI" at one end of the spectrum to "fully-CTI" at the other. At the non-CTI end of the spectrum there is a traditional ACD (with or without an IVR). At the full-CTI end of the spectrum is a closely bound-together configuration consisting of an ACD, an IVR, a computer system, and group of agent workstations, in which all telephony actions involve exchange of information/commands between the computer system and some or all of the other components. It is not easy to draw a line on this spectrum of possible arrangements that all experts would agree marks the boundary between non-CTI and CTI. For instance, even in a traditional call center arrangement, it is normal for the IVR to be linked to a computer system; but this in itself does not constitute CTI. The best approach to defining CTI is in terms of the capabilities of the arrangement.

In general, a CTI arrangement may be said to have been created if some or all of the following things are made possible:

- Information (such as an account number) **entered into the IVR** by the caller, or information (such as the caller's account status) looked up in a database using the IVR-captured information as a key, is transferred **from the computer system to the ACD**, and is used by the ACD to determine which group of agents (often called a "split") the call is routed to, when the call is transferred from the IVR to an agent. This may happen after the caller has entered all the information demanded up-front by the IVR (for example, an account number or a series of menu selection codes), or it may happen when the caller attempts to use the IVR to get information, but then gives up and "zeroes out" (that is, presses the "0" key) to speak to an agent.
- Information (such as an account number) **entered into the IVR** by the caller, or information (such as the caller's name) looked up in a database using the IVR-captured information as a key, is transferred **from the computer system to the workstation of the agent who answers the call** when the call is transferred from the IVR to the agent. This information is available at the point in time when the agent answers the call, typically in the form of a "screen pop" (a window that opens up on the agent's screen to display information about the new call).
- Information (such as the caller's account status), looked up in a database using, as a key, the caller's number – generally referred to as **the ANI** (Automatic Number Identification) or Caller ID – is transferred **from the computer system to the ACD**, and is used by the ACD to determine which split the call is routed to. Typically this arrangement would be used to direct calls to different splits, without requiring the caller to interact with an IVR before

speaking to an agent. This is particularly useful for "platinum" services, where the customer is considered too important to be subjected to the torture of an IVR.

- **The ANI**, or information (such as the caller's name and account number) looked up in a database using the ANI as a key, is automatically transferred **from the computer system to the workstation of the agent who answers the call** when the call is transferred from the IVR to the agent. This information is available at the point in time when the agent answers the call, typically in the form of a "screen pop".
- Information (such as the caller's account status), looked up in a database using **the ANI** as a key, is automatically transferred **from the computer system to the IVR**, and is used by the IVR to determine what choices are given to the caller.
- If the agent who first answered the call subsequently transfers the call to another agent, or back to the IVR, **the transfer actions are directed by commands, to the ACD, from the computer system or agent's workstation**. For example, the caller may be asked by the agent to select a new PIN. In order to keep this new PIN absolutely confidential, the agent transfers the caller back to the IVR by clicking on a "Customer New PIN Selection" button on the agent's screen. This sends a message to the computer system, which sends a message to the ACD telling it to transfer the call temporarily back to the IVR. The computer system then commands the IVR to play the "please enter your new PIN now" message. When the PIN is received and confirmed, the IVR passes it to the computer system. The computer system then issues a command to the ACD to transfer the call back to the agent. The computer sends a screen pop to the agent's workstation saying "New PIN selected OK".

In summary, a CTI arrangement can be said to exist when a computer system, working in combination with the agents' workstations, receives the caller's ANI and/or information entered into a IVR by the caller, processes that information, and sends information/commands that direct the flow of the call between the three telephony components – the IVR, ACD, and agents' telsets. In a sophisticated CTI arrangement the computer system maintains control over the actions of the ACD and IVR, even after the call is connected to the first agent who speaks with the caller.

Use of DNIS in CTI

In addition to passing the ANI to the ACD at the start of the call, the long distance carrier (or local service provider) may also pass, to the ACD, a string of digits that identify the number dialed by the caller to reach the call center. These digits are often referred to as "the DNIS". (DNIS is short for Dialed Number Identification Service.) In a non-CTI arrangement the DNIS is typically used by the ACD to steer the call to the right split (which is necessary where the physical paths from the carrier, such as T1 or PRI circuits, are bringing in traffic for more than one toll-free number).

Under a CTI arrangement, the DNIS may be used in combination with the ANI to carry out more "intelligent" steering of calls, and to provide agents with screen pops whose contents derive from the DNIS. For example, suppose that a call center handles traffic for both "gold" account customers and ordinary account customers, with separate toll-free numbers for these two classes of customer. If a gold customer accidentally dials the ordinary toll-free number, the computer system can make sure that the customer is routed to a gold agent, as follows. When the ANI is looked up in the database, and turns out to belong to a gold customer, even though the DNIS shows that the customer dialed the ordinary non-gold toll-free number, the computer system can command the ACD to place the call in the "gold agent" split. Furthermore, the computer can send

a special screen pop to the agent's workstation saying "Note: caller is gold but did not dial gold number. Please advise caller of gold number for future use."

Intelligent DNIS-handling is particularly important in call centers that support businesses which run advertisements for many products and special offers, each with its own toll-free number. For a returning customer (identified by his or her ANI, or by an account number entered into the IVR), information about which advertisement or catalog the caller is responding to*, in combination with the customer's account information*, can be used to provide screen pops that will help the agent handle the call more effectively. For example, by making use of the ANI plus the DNIS, the agent can be prompted to say "Good morning Mr Smith. You're calling about the special promotion on the inkjet printer, right?"

This is much more impressive than "This is the inkjet printer offer line. Could I have your name or account number, please?" By bringing together the two pieces of information – the DNIS that shows which number the caller dialed, and the caller's identity derived from the ANI (or from information the caller keyed into the IVR) – the CTI arrangement allows the agent to better serve the customer, and the agent's talk-time is reduced. The same screen pop can also suggest to the agent other products that the customer might be interested in, based on the product about which the call was made, *in combination with the customer's order history*.

TAPI versus TSAPI

Common use of the term "CTI" started in the early 1990s. At first, interest in the topic was combined with a great deal of confusion. At the June 1995 Nortel Users' Group Conference in Atlanta, "CTI" was a key session topic for the first time, and all the CTI sessions were oversubscribed. The attendees wanted to know "What is it?" and "How can we get some of it?" Confusion was probably at a peak in that year because two different "CTI standards" were being rolled out by vendors – TAPI and TSAPI. TSAPI (Telephony Services Application Programming Interface) was the result of a collaborative standards-definition effort by AT&T and Novell. TSAPI was subsequently embraced by Nortel and other PBX/ACD manufacturers. TAPI (Telephony Application Programming Interface) was the Microsoft standard, which Microsoft hoped would take over the telephony world.

As originally defined, TAPI was aimed at simply connecting a PC to a telephone instrument, so that the telephone could be made to dial a telephone number looked up in an application like Outlook. It also provided for the capture of the caller ID, from the caller ID display in the telephone instrument, passing the caller ID into the PC. TSAPI, on the other hand, addressed the need for sophisticated system-to-system links, between the ACD and/or IVR and a computer system. It allowed the computer system to: (a) capture information about a call, before the call is routed to a particular agent's telset; (b) transmit instructions to the ACD in order to steer the call to the right agent; and (c) manage the transfer of a call from one agent to another, or back to a IVR.

Since 1995, Microsoft has enhanced TAPI, and the differences between TAPI and TSAPI are diminishing. This makes the advice that most observers gave in 1995 – "forget TAPI" – less justified, while still adding to the potential for confusion in a CTI project today.

Good CTI implementations

These are some examples of good CTI implementations that I have come across or been actively involved with:

- In many of Citibank's overseas telephone banking call centers, CTI has been used to closely tie together the IVR, the computer system, the ACD, and the agent workstations. Calls are answered by an IVR. The customer enters an account number and PIN and can make basic enquiries (such as account balance). When he or she decides that it is necessary to speak to an agent, and presses "0", the call is transferred to an agent. When the agent answers the call, a screen pop shows the agent "the story so far": the customer's name, account number, and all actions performed on the IVR up to the point of pressing "0". This means that the agent can answer the call with, for instance, "Good morning Mr Smith. I see you were checking the balance due on your Visa card using our automated service. Do you have a question about that account?" This makes the customer feel that Citibank really has its act together: any bad feelings that the customer might have about being forced to use the IVR (rather than being connected to an agent from the start) are swept away when the agent greets him by name.

- In the same Citibank telephone banking centers, the agent can transfer the customer back to the IVR for a few seconds when the customer wants to select a new PIN, thus keeping knowledge of the new PIN secret even from the agent.

- When you get a new or replacement credit card from Citibank in the USA, as a security measure you cannot use it until you have made a call from your home telephone line to Citibank's call center to "activate" the card. When your call arrives at the ACD, the ACD passes your ANI to a computer system and connects you to an IVR. You enter your new card number. The IVR passes the card number to the computer system, where your account details are looked up in the database. If the ANI matches the record of your home telephone number in the database, your card is activated: you do not need to speak to an agent. Although this example does not involve the ACD transferring the call to an agent, I believe that it is accurate to describe it as a CTI arrangement. (By the way, I am sorry that these three examples are all from Citibank; but Citibank has really done well in these particular cases – although not others.)

- At some overseas Pizza Huts they have implemented a simple CTI arrangement to take the caller ID, look it up in a database, and (if the caller has ordered from that Pizza Hut before) provide the person handling the call with a screen pop with the caller's name, address, and order history. I was very impressed the second time I called one of these Pizza Huts and was greeted by "Good evening, Mr Hamer, would you like the same order as Monday night?"

To generalize from these examples, the following are characteristics of a good CTI arrangement:

- A good CTI arrangement is one that is good for both the caller and for the agent.

- A good CTI arrangement minimizes the number of pieces of information that the caller is asked to key into the IVR. (Ideally, when the caller is calling from his or her home or mobile telephone, the IVR simply asks "Are you calling from your home number?" – just in case the caller is calling from the house of a friend or neighbor who also has an account with the same company.)

- A good CTI arrangement provides the agent with all available relevant information about the call, and the caller, at the start of the call:
    - The customer's name
    - The customer's account number
    - The customer's address
    - The customer's telephone number(s)
    - What telephone number the customer dialed to get to the call center
    - What telephone the customer is calling from
    - What selections the customer made on the IVR, and/or what actions the customer performed on the IVR, before speaking to the agent
    - The customer's recent transaction/purchasing history
    - Recent enquiry or problem calls that the customer has made to the call center or helpdesk.

- A good CTI arrangement allows the agent to handle the call without ever touching the buttons on the telset. *All actions are performed using the keyboard of the agent's workstation*, including:
    - Signing on to a particular split at the start of a shift
    - Accepting a new call
    - Ending a call
    - Temporarily withdrawing from the split to handle administrative work
    - Transferring a call to another agent or call center
    - Connecting the caller to an external number
    - Signing off at the end of the shift.

- A good CTI arrangement allows the agent to transfer calls to other agents, along with the context of the call – so that the second agent gets a screen pop showing the caller's name, account details, and reason for which the call was passed to this second agent.

- A good CTI arrangement allows the agent to transfer calls temporarily back to the IVR so that the caller can key in sensitive information such as a PIN; and it will give the agent a screen pop showing the result, such as "valid PIN entered" or "new PIN successfully selected", without revealing the actual PIN.

CTI project failures

There are two categories of CTI project failure. The less common, but most clear-cut, is when the project manager goes to the business manager and declares that the desired solution cannot be delivered – because of constraints in the design of the existing components, the unwillingness of the various vendors to work together, lack of cooperation from the application programmers that support the core business application software, or some other reason along these lines.

The second, more common type of failure is where the CTI project is declared to be "completed" by the project manager, but where there are, in fact, critical parts of the intended CTI functionality that do not work. In this situation the various components have been tied together, to the extent that IP data packets can be exchanged between them; and at least some aspects of the arrangement work as planned. However, the things that the system as a whole can do fall far short of what was desired. (The business manager may have reluctantly accepted this situation. He or she may

hope that the missing parts of the functionality will be delivered at a later date, although disappointment is almost inevitable.)

For every successful and effective application of CTI there are dozens of failures. You can easily tell if you are being served by an infrastructure that represents at least a partial failure to achieve its CTI goals. These are some of the signs:

- The IVR asks you to key in your account number (or maintenance agreement number, home telephone number, or some other data to identify yourself). But when you finally get through to a human being, the first thing he or she says is "Can I please have your account number?" (Doesn't this make you want to scream?)

- In cases where the first agent that you speak to determines that he or she needs to transfer you to another agent, the second agent starts by asking you to give your account number yet again.

- Before you speak to an agent, the IVR asks you to key in your PIN, along with your account number; but before the agent will go ahead with a transaction, he or she asks you for your PIN again.

- You are calling from your home telephone number (and you do not have your caller ID blocked), but the IVR still asks you to key in the number you are calling from, without first asking the yes/no question: "Are you calling from your home number?"

- The IVR has indicated to you that it has captured your telephone number (either because you keyed it in or answered "yes" in the previous example); but when you are later talking to an agent, the agent has to ask you "What number can I call you back on if we get disconnected?"

- You have called a specific toll-free number for a specific service (such as Travel Emergencies); but the agent who answers the call asks you which service you are calling for.

The first one in the above list (asking the customer to repeat information already supplied to the IVR) is probably the most annoying and the most common. Most people would rather talk to a human being than an IVR. (Of course there are exceptions, where it is possible to get a very fast answer to a simple question from an IVR, such as a flight arrival time.) People's dislike of IVRs is intensified by poorly designed IVR scripts, with many menu levels, and too many choices at each level. The aim of the script designers is generally to minimize the proportion of calls that go through to an agent, because agent time is expensive and IVR time is cheap. Unfortunately, many script designers pursue this aim without counterbalancing it with some concern for the quality of the IVR experience from the customer's point of view. They hide the option to speak to an agent several levels down the menu "tree", in the hope that few callers will find it. Also, some script designers are just lazy. They cram seven, eight, nine, or even the full ten options (one for each key) into each question so that they only have to program a small number of menu levels; and they do not think about whether the flow of choices is logical. (I also suspect that a small, evil minority of IVR script designers derive pleasure from creating "the IVR from hell".)

Vendor-independent IVR research has shown that, on the whole, people prefer to answer a reasonable number of yes/no questions than a smaller number of multiple-choice questions. A well-designed yes/no script can be navigated faster than an equivalent multiple choice script,

because the earlier answers eliminate the need for the IVR to read out some of the later options. (Three levels of yes/no questions can cover the same choices as a single eight-choice menu.) With a yes/no script, frequent users of the IVR start to build a mental map of the tree of choices more quickly than with a multiple-choice script. If the IVR lets users "key ahead" of the IVR's questions, users more readily remember the pattern of yes/no answers needed to get to the most-frequently-used choices. For example, if the sequence to get an account balance is 112121, where 1=yes and 2=no, frequent users will key-ahead the sequence 112121 as soon as they hear the first question from the IVR. This use of key-ahead happens much less often with multiple-choice scripts.

In spite of these findings, script designers tend to opt for multiple-choice scripts, partly because the programmer is thinking of the questions as they appear on a screen (rather than as spoken questions), and partly because it takes a lot of hard work to design a good yes/no script.

This sad state of affairs in IVR scripting serves to intensify people's awareness of the other flaws in the call center infrastructure, bringing together the caller's frustration *and* the agent's lack of information about the caller at the moment the call is answered, and thus getting every call off to a bad start.

Why do so many CTI projects fail?

These are some of the common situations that represent pitfalls in CTI projects:

- **Pitfall #1: The project manager does not have skills that span telephony, data protocols, and application programming.** Compared with the number of engineers who are skilled at setting up IP networks and LANs, experts in voice systems are scarce. Those that stayed in the voice business (rather than "upgrading" their skills with some Cisco classes so as to be able to qualify for better-paying work on data networking and the Internet) tend to be purely voice systems experts, with little or no expertise in computer systems or application design. It is therefore very hard to find a project manager who is knowledgeable in telephony as well as data protocols and application programming. As a result, the critical element of CTI – *integration* – is lacking at the project management level, and individual experts in the project team operate in their own silos.

- **Pitfall #2: The project manager mistakes TSAPI or TAPI for the magic glue that is going to bind the components together.** TSAPI and TAPI are not solutions to a CTI problem. They are just standards used to establish the basic minimum level of inter-component communication. On top of TSAPI/TAPI has to be layered the necessary applications to make the desired things happen when one component talks to another.

- **Pitfall #3: The groups that maintain existing business applications do not have the resources to add the functionality that is needed to support the CTI functions.** In most pre-CTI situations an IVR has been introduced at the front-end of a call center. This IVR is introduced to the infrastructure by (a) connecting the IVR to an existing legacy application, and (b) writing scripts that run in the IVR. The IVR scripts combine basic language scripting with query-handling logic. This is how the early IVR vendors built their customer base. They provided the tools needed to get the IVR up and running with minimal changes in legacy business applications. This was a good thing for IVR vendors during the early days of IVRs; but it is no good for a CTI arrangement. To achieve CTI, *the computer system must become the control point for the call, not the slave of the IVR script.* This

means that significant changes and additions are needed to the application software that interacts with the IVR. The application software must interact with the other components (the ACD and the agents' workstations), as well as the IVR. Unless appropriate application programming resources can be made available to the CTI project manager, together with adequate testing time on a non-production copy of the database, no progress can be made on a CTI project.

- **Pitfall #4: The IVR is an NIVR**. It *should* be technically possible to use an NIVR in a CTI arrangement. However, CTI projects using NIVRs seem to fail more often than those using on-site IVRs. The problem with NIVRs is that they are configured by the carrier/telephone service provider to be a component that sits between the telephone network and the call center's ACD. Calls come into the NIVR, go through the "IVR phase" of the call, and are then handed off to the ACD, so that they can be put in queue for the next available agent. The NIVR is an unbypassable front-end to the ACD. By contrast, in a good CTI design, the computer system is the control point for the call, and the ACD is the central switching point: calls come in to the ACD, *then* go out to the IVR, and are later switched through to an agent – all under the direction of the computer system. Using an NIVR constrains the things that can be done with the CTI arrangement. For instance, it is not possible to send calls back to the IVR so that a customer can enter a PIN.

- **Pitfall #5: The vendors will not work with one another**. A CTI project is fertile ground for growing inter-vendor hostility and rampant finger-pointing. In a CTI project there are typically four technical groups involved in the project – the engineers of the ACD supplier, the engineers of the IVR supplier, the programmers working on the application software that runs on the computer system, and the programmers working on the agent workstation software. At least two of these are vendors; and in some projects all four may be vendors. There is no easy way to avoid finger-pointing when difficulties arise. Some project managers favor putting representatives of the four groups in a poorly ventilated room, with over-brewed coffee, until they agree on a plan to get the components inter-working properly. However, this may only serve to make future cooperation even less likely. In my experience it helps if the IVR supplier is also the ACD supplier: this at least reduces, by one, the number of external parties.

Avoiding these pitfalls

The above list is not an exhaustive list of pitfalls, but it covers the big ones. These pitfalls can be avoided as follows:

- Find the right project manager – someone with an over-arching understanding of the various technologies involved.

- Get the necessary application design and programming resources assigned to the project from the start.

- At the start of the project, clearly specify, in a detailed document, the desired end-to-end customer experience and the desired agent experience.

- Have as many people as possible review this document, including the business manager and some of the more experienced agents. (It is amazing how much you can learn from talking to the people who actually do the work.)

- Clearly document the target architecture of the system in a way that emphasizes the interactions between the computer system, the ACD, and the IVR.

- Don't use an NIVR.

- Draw signal-sequence diagrams to show the sequence of signals/messages that pass between the computer system, the ACD, the IVR, and the agents' workstations, on the various types of call that the call center will handle.

- During the project, conduct frequent walk-throughs of the interactions of all the components, to check that everyone (vendors and internal staff) is still working on their piece of the puzzle in accordance with the agreed overall architecture.