

Rossware Computing

# CyberOffice Handbook

*Providing the first and only modern, quasi-automated communicative link between a service-performing office and service consumer.*

What percent of work time, for personnel in your office, is spent in telephone communication with customers? Half? Perhaps more? Certainly, it's a lot. Consider the money saved, if you halved that. We believe, with systems described here, you can do that and much more.

How much time does a consumer spend, on average, in arranging for a needed repair, calling to inquire on status, working to re-schedule, so on and etc.? Is it five accumulated minutes per repair, ten, twenty? How much time is lost playing telephone tag? How great is the frustration when they can't take care of business unless they happen to connect during business hours? How much more would it mean, to them, if they could manage all such matters instantly, and at any hour wanted?

This "handbook" is provided to help you understand how to fully harness the system we've created to achieve these purposes. If you follow the prescriptions here, we don't think you can help but reap rich rewards.

## General Description

The main concept is to move from a person-to-person, voice-based mode of communication when coordinating work with your customers, to one based on email and the internet (for a highly practical overview in which you can actually see and work in the forms involved, go to our website <http://rossware.net>. In the left-hand column, click on 'CyberOffice.'

## Outline of Scenarios

1. Initial job creation and scheduling (via interface provided on your website).
2. Initial scheduling on an existing order -- where the order is received from a third-party, such as a home warranty company, landlord/tenant situation, or similar (ServiceDesk emails the consumer a request to schedule via a '*click here*' hyperlink, which takes them to a venue on your website).
3. Re-scheduling on an existing job after parts arrive, or similar (again, ServiceDesk emails the consumer with a request to schedule via a '*click here*' hyperlink, which again takes them to a perfectly-designed interface on your website).

4. Reminding/Confirming on tomorrow's appointments (ServiceDesk emails customers, reminding them of the appointment, providing the tech's ETA, and providing a '*click here*' hyperlink, via which they are again taken to a perfect interface on your website, this one allowing them either to confirm, or to reschedule if required).
5. Checking on job status (customers can go on-line to see the status of their jobs).
6. Technician tracking (a variation of "tracking a shipment," our version allows the consumer to see when the tech left the office, how many jobs are on his roster, where the consumer fits on his list, and where he is on his way working through it).

## The Basic Scheme

The primary operative elements are: (1) a web-interface that's hosted on a remote server (somewhere in Texas, we think); (2) a compact utility that runs in the background on one station (pick whichever one you prefer) in your office; (3) ServiceDesk; (4) your own website; and (5) email.

Rossware leases the remote server, and there is no need for you to worry about the details. It's actually the remote server that hosts each of the applicable on-line interfaces, and which holds the data that allows them to interact intelligently with your consumer. Thus, when in the last section we said those interfaces are *on your website*, we lied. The intent is that, while it should *appear* to your consumer as *though* they are on (or in) your website, the actual machinery is on that remote server. This eliminates any need for you to install, host and manage the machinery yourself. Overall, it makes the whole thing much easier.

The compact utility is called "*SD-CyberLink*" (CyberLink for short). It's a tiny program whose job is to periodically upload information to the remote server. This is information that allows the various interfaces (as provided there) to interact appropriately with your customer. In particular, it will initially upload information about your company and setup (contact telephone number, email address, etc). On a periodic basis (typically once per 10 minutes, though you are free to change the interval), it uploads live data that indicates the days on which you still have vacancies for scheduling within various portions of your service territory (so as to allow the interfaces to know what dates they can offer for appointments).

In addition to *uploading* information, it's also the CyberLink's job to *download* information from the remote server, whenever relevant information appears there. When a new job is created and booked via Scenario 1, for example, CyberLink grabs the info and plugs it all appropriately into ServiceDesk. When a customer schedules in response

to a Scenario 2 or 3 email, that info is likewise grabbed by CyberLink and plugged into ServiceDesk, as is (indeed) confirmation (or rescheduling) data when the customer responds on a Scenario 4 request.

Besides being the recipient of on-line job-creation, scheduling and confirmation activity (as downloaded by CyberLink), ServiceDesk is also the entity via which you'll generate and send the email requests that initiate Scenario 2, 3 and 4 activities. For each Scenario, there are contexts within ServiceDesk for initiating these setups, as will be further discussed below.

The role of your website is to provide a context for the on-line interface with which your consumer interacts. The general design intent (again) is even though that interface is served and operated via the remote server, it should appear to your consumer as though they are doing everything right on (or within) your own website.

Email is, simply, the mechanism by which you'll be communicating to your consumer that: (a) she needs to connect for scheduling; (b) her parts have arrived, and she needs to connect for rescheduling; or (c) her appointment is tomorrow, and she needs to connect to confirm or reschedule. Email is the primary basis of *outward* communication, in other words, from you to the consumer.

The on-line interfaces, by contrast (for Scenarios 1 through 4, at least), are primarily a means of *inward* communication, from the consumer to you (Scenarios 5 and 6 being the exception—since, like emails, they are outward communication to the consumer).

## **Setting up SD-CyberLink**

The CyberLink utility will be provided to you via email, with installation instructions included in the mail.

We do want to emphasize the importance of one detail. Please be sure to follow the advice (as provided in that email) to setup Windows so it will auto-start CyberLink (at the computer where you've determined to run it) when Windows itself boots.

Otherwise, humans being what we are, there is every likelihood you'll forget to re-start CyberLink after some Windows rebooting event, and for some indeterminate period you'll have customers going on-line to schedule themselves, with no prompt acknowledgement being sent (not a good impression to give your consumer), no information popping into ServiceDesk in response, and so on.

For obvious reasons, it's important for that utility to run 24/7.

## Setting up the ServiceDesk ZoneScheduler

If you've previously setup for reception of automated dispatches via ServiceBench, ServicePower or LG (using any of the applicable *DispatchLink* utilities), it follows you already know about (and have implemented) the ZoneScheduler system within ServiceDesk. In such a case, there's no need to read further in this section. CyberLink uses the very same mechanisms to determine your availability status for uploading to the remote server (which allows it, then, to present available scheduling dates to your on-line customers).

If you have not (by reason of such prior use) already had occasion to acquaint yourself with ServiceDesk's ZoneScheduler system, it's time to do so now.

The basic concept is to provide a venue in which you can indicate what your job capacity is for given days. Based on this, utilities such as those DispatchLink utilities and CyberLink can make a comparison, for any given day, between indicated capacity and what's actually scheduled—in order to determine if vacancies remain.

An added twist is that, if beneficial to overall strategy, you are permitted to divide your territory into as many "zones" as wanted, each being defined via a list of zipcodes. This is particularly helpful if there are areas you wish to service only on particular days of the week (in which case you could make their allocations zero on other days), or if your territory is so large that it's impractical, say, to have techs that normally work in one region cross-over and help out in another region on days when their area is light and the other heavy.

At any rate, if you've not done so previously, you'll need to setup the ZoneScheduler system. We've long had an instruction document for the purpose. It's called *ZoneSchedulerInstructions.pdf*, and can be found in the c:\sd folder at any station where ServiceDesk has been installed (look in the root \sd folder of the server drive if you're setup in thin-client mode).

## Dealing with your Types, Makes and Dealers Lists

When you examine the CyberLink utility's interface, you'll see it has a button labeled '*Upload Core Values.*' That button's function is to upload, to the remote server, several details about your company, including the zipcodes you service. Importantly, it also uploads, from the ServiceDesk UnitInfo system, its lists of machine Types, Makes and Selling Dealers.

The web-scheduling interface needs these lists, so it can present them as dropdowns, to your customer, when she schedules on-line.

Given that such lists will be presented to your on-line customers, it's a good idea to make sure you've got them cleaned up and optimized, within ServiceDesk, so your customers will see exactly what you want them to.

And then there's option 2.

We've ran into a couple of users who wanted to maintain very extensive lists within ServiceDesk, while presenting more limited lists to the on-line customer. If you'd like to do this, simply make a new copy of the *UnitInfo.mdb* file (as found in the *lsd\nedtata* folder on your server). Name this new file *UnitInfo-ForWeb.mdb*, and place it in the very same folder. Open this new file in Microsoft Access, remove all the tables except the three of concern (*DealerList*, *MakesList* and *TypesList*). Then edit the lists to preference.

As for how the CyberLink utility reacts, it simply looks (in all cases) to see if you've provided this alternative UnitInfo file (i.e., the one named *UnitInfo-ForWeb.mdb*). If it sees that you have one, it pulls the lists (for uploading to the remote server) from there. If not, it pulls from your standard UnitInfo file.

## Details for Operating Scenario 1

Aside from the matters above-discussed, you must do two things:

1. Add an appropriate *hyperlink* to your website.

Basically, you'll have a button (or something similar, even just an area of text, if preferred) that advertises the ability to schedule on-line. It will be configured so that it's apparent to the user they should click on the object to fulfill the purpose. The object, in turn, is configured with what's called a "hyperlink," which is simply a particularly-configured reference to another webpage.

When Rossware sets you up, they'll provide the exact hyperlink that's needed for your company. It will be a variation of the following:

<http://sched.rossware.net/?id=1001>

In fact, it will be precisely the above except for the final number (i.e., the "1001"). That portion tells the on-line server which company the consumer is connecting for. It allows the remote server and interface to know it's working for you, and to interact with the consumer on that basis.

Whenever a webpage object is setup with a hyperlink, there are options as to how it's configured. Specifically, it can be configured

so that when the referenced page opens, it does so as: (a) a new page in your browser (i.e., the old page stays and is not replaced; (b) a replacement to the existing page; or (c) a new display *within a designated frame or page*.

You obviously may configure your setup according to whatever most pleases you, but our recommendation is to go with option C. This creates the definite impression, for your customer, that she is doing everything right *on* your site. Plus she'll continue, as she works, to see your carefully-designed logo and graphics in areas surrounding, so on and etc.—making for an altogether better marketing effect.

2. Configure the station where CyberLink is running for successful emailing (this is needed for all of Scenarios 1 through 4).

When CyberLink downloads any new appointment (whether it's Scenarios 1, 2, 3 or a rescheduled 4), it immediately emails the consumer confirming that you've received the appointment (there are other helpful pleasantries, as well). At least, it *tries*.

We say "tries" because CyberLink cannot send any email on its own. It relies on you having an email system setup within Windows and properly configured for the purpose.

Specifically, you must have a MAPI-compliant email program properly installed, specified as the Windows "default" email program, and setup with an email "account" via which it is able to successfully send out emails.

If you've been using Hotmail, Gmail, Yahoo or anything similar, please understand these are a completely different animal than is an *installed* email program. With all of those, you're actually going on-line to someone's website to manage your mail. With an installed program, by contrast, you're doing the work right on your local machine. The local program downloads emails from your provider, where you then work on them, and at that point it doesn't matter whether you're connected or not. Similarly, when you create an email, all the typing and stuff is done locally. A connection then matters only when your program goes to upload the mail to your provider.

Also, if you've formerly been using web-based mail, don't worry about any cost as connected with setting up an installed system. Virtually all internet service plans include several "free" email accounts. In other words, you're already paying for some quantity of *installed* email accounts, and have just never yet used them.

As for the installed program of choice, in almost every instance the best solution will prove to be none other than Microsoft's *Outlook Express*. This is the program that installs by default in every Windows computer (in other words, you've paid for it already as well). You can try other programs, but it's our experience you'll encounter frustration (including with Outlook, which is Microsoft's beefier and premium, i.e., it-doesn't-come-with-Windows, email program).

Also, we see no purpose in *attempting* to use a different program. No matter what you may otherwise prefer for managing the emails in which you're actively working (i.e., reading with your own eyeballs, typing new emails to send out, etc.), making Outlook Express the *default* email program will not in the least prevent you from using the other program you've preferred. You may continue with it just as you always have.

If you have no idea how to setup Outlook Express (and an email account within it), our suggestion is to call the HelpLine at your ISP (stands for Internet Service Provider; it's the company that takes money from you every month in exchange for providing internet access). They should be able to walk you through it in short order. They'll also know certain connection codes that are essential to successful setup.

There are three significant imperatives once you get Outlook Express installed and able to successfully send emails.

- (a) Tell Windows that Outlook Express is the "default" email program.

This is actually done from Internet Explorer. Open that program, click on 'Tools,' then 'Internet Options,' then the 'Programs' tab. In the 'E-mail' box, select 'Outlook Express,' and click 'OK.'

- (b) Tell Outlook Express to refrain from pestering you when a program tries to send emails.

One of the strategies used by ambitious viruses, as they try to replicate themselves, is to email themselves to other computers, using your email program. As one measure against this, Microsoft has Outlook Express programmed to detect when an email was initiated by other than direct user action, in which case it asks for direct user approval before letting the email go out. That protection is a fatal impediment to CyberLink scheduler sending out those

confirmatory emails in a timely fashion (what if you're not there at midnight when some customer goes on-line to schedule?), not to mention, you hardly want to be pestered every time an email needs to be sent.

For such reason, the protection must be turned off.

To do this, in Outlook Express, click on 'Tools,' then 'Options,' then the 'Security' tab. Look for the box labeled '*Warn me when other applications try to send mail as me,*' and uncheck it.

- (c) Re-start Windows. The above changes do not otherwise seem to take.

Just so you know (and to emphasize), CyberLink does not handle the direct processes as involved in sending emails (nor does ServiceDesk, or any other Rossware product). Instead, all Rossware products rely on Windows for the purpose. Essentially, they call over to Windows with a statement that (in layman's terms) is akin to saying: "Hey, Windows, please take this email and send it for me." Really, that's as deep as our programs get into the emailing machinery. We explain this so you'll understand, in most cases where there's an emailing issue, it will be something you need to work out in Windows and/or its default email setup. We may be able to help, but the fault is likely there, and not in the Rossware product you're using.

At any rate, with the above elements setup (and CyberLink properly configured, running, and with its 'Core Data' having been uploaded), you're now ready, cocked and primed to begin seeing new, scheduled service requests pop right into ServiceDesk—wondrously, without any office personnel having expended even a moment speaking with your customer.

## **Details for Operating Scenario 2**

Suppose a property management company calls, seeking service at one of their rentals. Or, perhaps it's a home warranty company seeking service for a policy holder. Regardless, they provide all the order information, and you now need to contact the ultimate consumer, for scheduling.

If not using on-line scheduling, you might very well keep the order information in a ServiceDesk Callsheet, pending success in reaching the consumer and creating the first appointment (at which point, you'd then do the Job/Sale process).

For the purpose of using Scenario 2, you're going to change the sequence. The reason is, the on-line scheduling interface needs a JobRecord with which to connect the consumer's on-line work (the associated JobRecord/Invoice Number provides a unique connecting key).

So, don't wait. Go ahead. Right away, when you get the order, do a Job/Sale operation from the Callsheet. Then, go to the resulting JobRecord, and there invoke the process to email a scheduling request to the consumer (this presupposes, of course, that you've been provided an email address; otherwise you'll be stuck with conventional methods).

The JobsCurrent form in ServiceDesk suffers a problem. There are more things that can be done there than there is space for unique buttons. For that reason, we're increasingly making single buttons do double-duty. That is now the case with the '*Scheduling*' button. Formerly, it had one function: you could left-click on it (or strike Alt-S on your keyboard) to initiate the standard scheduling options.

Those functions are still exactly the same, but now there is a second option. It's invoked by either right-clicking on the button (as opposed to left-clicking) or striking Ctrl-S on your keyboard (as opposed to Alt-S).

Specifically, this alternative invokes the process where ServiceDesk sends an email to the consumer, asking her to click on a hyperlink (i.e., within the body of the email) to schedule herself. ServiceDesk simultaneously uploads applicable data to the remote server, so that when your consumer connects via the email-provided hyperlink, it will have the means with which to intelligently interact. Naturally the event (of you having emailed the scheduling request) is auto-recorded in the applicable JobHistory, and the whole process consumes perhaps a second.

When the consumer receives the email and proceeds as directed, she's taken to a very nice interface where the scheduling task is performed. Within moments, the resulting information will pop perfectly into ServiceDesk (via action of the CyberLink). In short, you'll see the appointment appear in the ScheduleList and DispatchMap, and the narrative JobHistory will be appended to explain who went on-line (and when) to schedule the appointment.

### **Details for Operating Scenario 3**

This is much like Scenario 2, except the intent is for use on jobs where the tech has already been there, ordered parts, the parts have arrived, and re-scheduling is now needed.

The emailed re-scheduling requests can be invoked from either of two contexts:

- (a) When you check-in the last of any/all parts as ordered on a job, ServiceDesk will automatically ask if you want it to email the request; and
- (b) You can *volitionally* initiate the request, from the JobRecord, using exactly the same method as described for Scenario 2 (i.e., from the JobRecord, either right-click on the 'Scheduling' button or strike Ctrl-S on your keyboard).

In this last context, incidentally, ServiceDesk detects (based on information in the JobHistory) that you're in a Scenario 3 as opposed to Scenario 2 context, and behaves accordingly.

Everything else, from the perspective of what you're doing within ServiceDesk, is exactly the same as in Scenario 2 (ServiceDesk itself configures the emails appropriately to the circumstance, uploads applicable data to the remote server, etc.).

#### **Details for Operating Scenario 4**

The notion here is, sometime in the afternoon or evening of each day, you've worked out the assignments and sequence of jobs for tomorrow's work (if you're not using the auto-time-frame-estimator for this purpose, we highly recommend it). Now you need to: (a) remind the customers of their appointments; (b) confirm they'll actually be there; and (c) inform them of the time frame within which you're expecting the tech to arrive.

Obviously, if done via manual means, the above is a very time-consuming and laborious task.

To automate the process, display your ServiceDesk DispatchMap, then strike Alt-P on your keyboard (this is the command that has traditionally given the general print options, hence the 'P'). Select the option labeled '*invoke Dispatch options, but for all techs,*' then (in the next display) the one labeled '*email a confirmation/reminder to each customer.*' Finally (and in the next display), pick either of the two options that indicate your customer will respond via a website hyperlink (in this and some other contexts, you have the option to view the email before it goes, versus having it just go without further bother).

In this manner, you can send out scores or even hundreds of confirmation requests, with a time investment (on your end) of mere seconds.

Don't worry if you don't have email addresses for everyone; for those lacking, ServiceDesk will simply refrain from attempting the email, plus give you a list of those for whom this was the case (thereby prompting you to use more old-fashioned means in their regard). In addition, symbols in

the DispatchMap will change to distinguish those for whom requests have gone out as compared to those for whom they have not.

The expectation is that most of your customers will receive the reminder/confirmation requests, respond by clicking on the included links, go on-line thereby, and confirm their appointments. Via downloading by CyberLink, that info will then pop right into the DispatchMap (where symbols will change to show the appointment was confirmed), and into the narrative Job Histories (which will indicate when the appointment was confirmed and by whom).

If any of your customers elect to change their appointments, that fact will be correctly noted in all appropriate contexts as well, plus CyberLink will create a Callsheet, further calling to your attention the unusual circumstance.

The next morning, a mere glance at your DispatchMap will rapidly divulge any holdouts (i.e., folks who still have not confirmed). It's then a simple matter (with that remaining small minority), to call and see if you can confirm via old-fashioned means.

### **Details for Operating Scenario 5**

Job Status Checking (i.e., letting the consumer go on-line to determine the status of their job), is actually a capability we've had for some five years. However, it's been based on a set of mechanisms that have nothing to do with machinery described here.

If you're anxious for the ability, you can use that old set of mechanisms right now. However, we are planning to build a better capability, in line with and employing the same underlying mechanisms as are above described. This will be a better capability, easier to implement, and so on. However, we don't actually have it just yet.

### **Details for Operating Scenario 6**

Technician Tracking is an idea we had long after work was well under way on Scenarios 1 through 4 (in fact, converting Scenario 5 capability to this new set of mechanisms was also a late-conceived idea). Anyway, implementation of this item, as well, is not yet in place.

### **Optimizing Presentation of the On-Line Interface, Scenarios 2 through 4**

As mentioned when describing how to setup a hyperlink for Scenario 1 (pg 5), for that context you control how the interface is presented (within what frame, whether on a new page, etc.) by virtue of properties as attached to the link itself, which actually is (physically) on your web page. Notably, that control method is not available for several of the other interfaces, for

(rather than invoking from a hyperlink on your web page) their on-line presentation is invoked via hyperlinks in an email.

Certainly, you could allow these other interfaces to present themselves to your customer naked—meaning, *without* surrounding text and/or graphics, of your design, promoting your business. We suspect, however, you'd prefer some decoration. In particular, you'll likely want them to be presented as *pages within* your website, even though they're invoked from mere a email that does not even reference the latter.

To accomplish this involves three easy steps:

1. Add a page to your website, within which you want Scenarios 2 through 4 interfaces to appear.
2. Within that page's underlying html code, paste the text string that's provided in the document called 'StringForInsertionToContainerPage.txt' You can download this from <http://rossware.net/downloadSDWS>).
3. In CyberLink, there's a box labeled 'Frame url.' Within that box, type the url for the page described in Step 2. Then click on the 'Upload core values' button, so this url can be uploaded to the remote server.

That's it. You should be able to test, now (from ServiceDesk, create a dummy emailed request on any of Scenarios 2 through 4, then click on the hyperlink in the resulting email), to verify that the on-line interface opens within your own webpage, as intended.

## **Customizing Text as Presented to the Consumer**

In several contexts, this system presents the consumer with communicative text: sentences and paragraphs that are designed to convey important elements of meaning. We've done our best to pre-"can" this text for wide acceptance and optimum clarity.

However, we recognize you might want the text presented differently. Thus, we've configured the system to allow customization in this regard (for any situation, by the way, you can see and review the canned text by faking the scenario).

1. Scenario 1 and 2, Acknowledgement of Initial Appointment Made On-Line, as Sent by CyberLink:

To create a custom email for this context, type it in any text editor, and save it to a file called *MyTextForConfirmingEmail.TXT*. Save the file in the *lsd\inetdata* folder on your server. There is just one caveat: at the place in your text where you want the applicable appointment

recited, place the text “[AppmntDate]” (include the brackets and all; they’re a critical part of it).

2. Scenario 3, Acknowledgement of Re-Scheduled Appointment Made On-Line, as Sent by CyberLink:

Type the wanted text. In this case, save to the same folder location as above described, but name the file *MyTextForConfirmingEmail-AftrFrstAppmnt.TXT*. Use the same method as above-described to indicate where, within your text, you want the appointment recited.

3. Scenario 2, Request to Schedule, as sent by ServiceDesk:

Customization not provided presently. If you need it, let us know.

4. Scenario 3, Request to Re-Schedule, as sent by ServiceDesk:

Customization not provided presently. If you need it, let us know.

5. Scenario 4, Request to Confirm, as sent by ServiceDesk:

Customization not provided presently. If you need it, let us know.

## **Enhancing Collection of Email Addresses**

It’s quite obvious that much of the automation, as here discussed, depends on having your customer’s email address. For customers who’ve initiated their jobs on-line, that’s not an issue (the email address comes, part and parcel, with the job).

For customer’s who call conventionally, by contrast (i.e., via telephone), your call-takers are going to have to ask for a new item of information. You’ll have to overcome some inertia, in this regard, because, obviously, your call-takers are not used to doing this, and we all know human nature. Certainly, you could institute some reward program to encourage call-takers to change their habits, but our thinking is that the most important factor is assuring they are trained to ask for the email address in a manner that keeps both they and the customer comfortable. Much of that effort, we believe, is teaching them how to use optimum dialog.

For example, one way of requesting the email address would be as follows:

“Just so you know, Mrs. Jones, our primary means of communicating with you during the course of this repair will be via email, so I’ll need that address too, please.”

You should teach your call-takers that, with so simple a statement, the consumer is simultaneously assured that you have a valid need for her email, and she's put on notice that she should be checking it while the repair is pending.

An alternative dialog, after having asked for and received regular address and telephone numbers, could be:

"I also you need your email address. The reason is we'll use it to keep you apprised of what's going on with your repair. We'll even email to remind you of the appointment the night before."

A little role-playing and rehearsal with your call-takers is likely all you'll need to get them going. Maybe even make a party of it, with refreshments, and so on, to make it worthwhile and fun.

You likely also should include a little practice as to how to react when a customer says they have no email. Our thinking is a simple reaction such as:

"Very well then, we'll just have to communicate with you the old-fashioned way."

should do nicely.

Don't underestimate the effectiveness of role-playing, especially in a group setting. It may seem corny, but it works. On top of that, if you have some kind of special meeting for the purpose, it will imprint far more indelibly in your call-takers' minds that, henceforth, they really will be doing their jobs differently in this respect.

One more suggestion is that you role-play how to give assurances when a customer is worried that her email address might be used for marketing. Actually, you first need to decide what your policy will be in that regard. Most obviously, you'll never share the email address with others, and that can be one level of assurance to the customer. But an intermediate question is whether you'll allow yourself to use the email address for your own marketing. Our recommendation is no, for this allows your call-takers to assure the customer that the email will *only* be used for managing the job. But you need to decide (and inform your call-takers), so they'll know just how much they can honestly promise. Be sure you rehearse with them, so they'll know how to do it optimally.

Our estimation is that, if you follow suggestions as outlined here, you'll be able to acquire email addresses in at least 80 percent of the cases where consumers phone in seeking service. That's now, in 2007. In a very few years, we figure, it should get extremely close to 100 percent.

Of course, today that is only one method of receiving job orders. In particular, you may have a significant percentage of orders coming from home warranty companies (hopefully, automated via our *EmailedDispatchReceiver*). You may have another percentage coming directly from manufacturers (hopefully, automated via our *DispatchLink* utilities).

Regardless of the method, it will obviously be strongly to your advantage if these third-party vendors of service provide the consumer's email address, as an added element in the information conveyed to you. For any that do not do so presently, our suggestion is that you beg, plead, kick and cajole—seeking to persuade them to do so.

Remember, yours is not the only voice. If you're pestering a particular entity (like, say, American Home Shield), and so are several of your compatriots, there's a hope (at least) that eventually such voices will be heard (if you're curious, at Rosswire we've already begun the campaign).

### **Suggestions for Enhanced Marketing**

To the consumer, scheduling on-line can be a huge convenience and attraction. For that reason, we suggest promoting the concept broadly. Advertise the ability (with appropriate url) on your business cards and invoices, in your yellow pages ads, on stickers that are attached to the machine that's been repaired, on promo magnets, and so on.

On your website directly, we suggest making the ability to directly book *very prominent*. It's a big sales feature, after all. In fact, if you have one of the fancy websites with several different pages, it's likely a good idea to have a big prominent button on *every single page*.

We also suggest modifying your recorded voice greetings, to inform consumers that instead of waiting for your return call or in the queue, they can go on-line *immediately* to schedule their repair. No doubt, a good many consumers will say "Yea, that's even better." They'll hang up and do it post haste—thus assuring you actually get the job (i.e., they won't call someone else after losing patience when waiting to communicate with you). Plus, your queue is trimmed for others.

You might also train your techs to casually ask the consumer, as they're engaged in conversation, if the job was booked on-line. If the answer is no, that will lead (quite naturally) to discussion of the fact that it was an option.

All of this should further distinguish you, and give you a significant edge over the competition.

Beyond direct-to-the-consumer promotion, don't forget those who order service (or might *like* to do so) on behalf of consumers. Property managers are one strong possibility (how much more would they like to be able just to go on-line and order service?).

Most particularly, what about those local dealers that you'd like to have always send work your way, instead of to a competitor? Go there. Show the sales personnel how easily and immediately they can book a job for their customer. Sales people *want* to be heroes. Give them the means by which to do so, and they'll love you for it. More importantly, you can then all but count on the work being yours.

### **Other Surprising Uses:**

One day Karie and I, here in the office, were surprised to learn that an early adopter of on-line scheduling was using it for a purpose we'd not thought of. He employs a couple of gals that handle incoming calls from their homes. Instead of bothering to network them into his office, he simply has them go on-line to his scheduling page, and book calls in that manner.

It's possible you could have a commercial answering service book calls similarly. We're not sure of all the potential variations, but we're betting that, with the imagination of all you clever folks out there, we'll hear of still more cool ideas. Please let us know.