

Rossware Computing

CyberOffice Handbook

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

What percent of work time in your office is spent on the telephone with customers? Half? Perhaps more? Certainly, it's a lot. Consider how much you'd save if that time was cut in half, while even better managing the same work flow. 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, etc.? Is it five accumulated minutes per repair, ten, twenty? How much time is lost playing telephone tag? How great is the frustration when the customer can't take care of business unless they manage to call 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 (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 set of specialized web-interfaces that are “plugged into” your website from their source on a remote server; (2) a compact utility that runs in the background on one station in your office; (3) ServiceDesk; (4) your own website; and (5) email.

Those specialized “plug-in” interfaces are in fact “hosted” on a Rossware server. Thus, it will only “appear” to your customer that, when within these interfaces, they are directly “on” your website. Technically, they’re on interfaces provided by Rossware (though fit into windows within yours). The strategy eliminates any need for you to install, host and manage the machinery yourself, and overall makes the setup 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 that 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, and other data.

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.

As for entities that send emails, two other applications (i.e., aside from ServiceDesk) also get in on the act. CyberLink itself is configured to send emails after a customer confirms an appointment, with an invitation and hot-link that makes it easy for them to use the tech-tracking feature (assuming, of course, you've opted to use that element). And the SD-MobileLink program is configured to optionally send emails, when a technician has performed a visit and yet not completed the repair, with an invitation and hot-link that makes it easy for the customer to track ongoing job-status.

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.

To summarize, the CyberOffice system consists of outward communication and inward.

Outward communication consists, within applicable web-interfaces, of such things as availability for particular days scheduling as applicable to particular zips, job status and technician tracking. It is also provided via a series of different emails as generated by ServiceDesk, CyberLink and MobileLink.

Inward communication consists of info generated by your customer when she interacts via the online interfaces, including her acts when: (a) initially scheduling; (b) re-scheduling after parts have arrived; and (c) confirming (or re-booking) her appointment.

All functions from your end are automated or, at least, semi-automated.

Setting up SD-CyberLink

The CyberLink installer may be downloaded from CyberOffice downloads page on our website (<http://rossware.net/downloads/SDCO>). It runs like most any other Windows installer. We do not update the installer often, which means more than likely it will be installing an older copy of the program. For such reason, we suggest you perform an update immediately after doing the install.

We also want to emphasize the importance of one detail. Please be sure to follow the advice (as provided in your introductory 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 is likely no need to read further in this section.¹ 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

¹ The exception would be if your ZoneScheduler setup is configured to use *just one zone*, and is also using the option that allows you to forego creation of a ZoneList.txt file (with this option, all related functions assume every job fits within your one [default] zone). CyberOffice cannot directly rely on this mode, because it needs an actual list of zips, to upload to the CyberOffice server (in turn, this allows the on-line scheduling interface to know if a consumer with a given zip is a legitimate candidate for scheduling).

Given this need, you must at minimum provide CyberLink with a list of zips. To state it otherwise, even if you've otherwise setup the ZoneScheduler system, but have chosen the "*no-ZoneList-file,-default-zone-only*" mode, you'll in fact have some zone-setup work to do for CyberLink.

If, in particular, you want to stick with that mode (whether already setup that way for other purposes, or setting up presently), you can satisfy CyberLink's need for a list of zips by making a different file.

Specifically, please make and save to your server's \sd\netdata folder a file called **ZipsForCyber.txt**. For contents, simply place in the file one line of text for each zip you want to offer for service within CyberOffice. Each such line should consist simply of said zip, and nothing else.

Again, if you've setup ZoneScheduling in ServiceDesk for other purposes (and without relying on the "*no-ZoneList-file*" mode), there is no need to even be concerned with this. The simple bottom line is that the CyberLink program needs a list of zips, one way or the other.

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 \sd\netdata 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,² 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.

² If you don't possess Microsoft's Access program, it's easy to workaround. When you make that copy of the existing *UnitInfo.mdb* file, save it (and this is prior to re-naming it) to the \sd\netdata folder at a station in your network (but not the server itself) that's operating in Thick-Client mode (i.e., it has its own set of \sd folders). Then, at that station and from within ServiceDesk's *Settings* form, momentarily set it to use its own c:\ drive as the server. At this point, ServiceDesk (at such station) should be using other than its normal operating files (i.e., what you've placed into its local c:\sd\netdata folder, including your to-be-edited copy of the *UnitInfo.mdb* file). From this instance of ServiceDesk, go into the UnitInfo form and use its mechanisms to edit the lists as desired for website presentation. Don't worry about otherwise messing up your UnitInfo data here, for this is only a copy you're working with. After you've edited the lists as desired, re-set ServiceDesk to use the proper/intended drive as server (i.e., so you're back to the proper operational setup, looking at real operating data). Now, go to the copy you made and edited (i.e., via ServiceDesk in temporary/not-true-server mode). Re-name it (per above instructions), then copy it into your true server's \sd\netdata folder (note that you're not replacing the operating file; you're simply moving in a new one under different name).

If you don't have any stations operating in Thick-Client mode (i.e., all are setup as Thin), another workaround (this would need to be done during non-business hours, when others -- including techs in the field via SD-Mobile -- are not otherwise accessing the data) would be to make a copy of *UnitInfo.mdb* that's intended as your "this-is-my-real-copy-for-true-operation" instance. Save it elsewhere (i.e., a place that's safe from editing, outside the \sd\netdata folder). Then use the UnitInfo form from within ServiceDesk to edit the lists as wanted for web-presentation purposes (again, don't worry about what you're otherwise doing to the data). After you've finished, go into your server's \sd\netdata folder and do the above-specified re-naming on what was formerly the operating *UnitInfo.mdb* file. Then, go back to the copy you made at the beginning (i.e., your "this-was-my-real-copy-for-operation" instance) and copy it back into your server's \sd\netdata folder.

You are, in short, doing a simple dosie-doe, and it's easy if you simply think it through. Be sure if using the second method, though, that you *don't fail to copy your operating copy of the file back into the server's \sd\netdata folder*. That last step is **critical!**

Details for Operating Scenario 1 (Initial Order and Booking of Service)

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).

Since Cyberlink will be conducting a lot of business for you via email, it's imperative that the computer where it's running is properly setup for emailing purposes. In principle, this is very easy. In practice, it can also be very easy. However, there are potential pitfalls, and it's an area where, if you're equipped with a somewhat larger understanding, you'll be much more comfortable and competent to assure your setup is as it needs to be.

This area of concern is sufficiently large to merit a few pages of explanation (enough to assure you really understand the concept). For that reason, we have a separate tiny document to assist you:

<http://rossware.net/MiniManuals/How%20to%20Setup%20Email.pdf>

If you don't already have a clear understanding of how email integration needs to work (between programs running in your computer and the email system itself), please open and read the above. It is simple, non-technical, and the understanding it conveys will make you better equipped to handle a variety of future situations.

With each of 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 (Email-Link-Initiated Scheduling of First Appointment After Separate Service Request)

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 (Email-Link-Initiated Scheduling of New Appointment after Parts Arrive)

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 (Confirming Appointments)

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)

To let the consumer determine the status of their job, online, is a capability we had many years prior to introducing CyberOffice. It was done via a stand-alone utility called the *WebPageUpdater*, combined with a bit of PHP code we provided for your website. But that was a first-generation capability. Via CyberOffice, we've re-introduced essentially the same function, but packaged in a fully modern context.

Your first step in implementing this feature is to setup your website to accommodate the status-checking interface. To do so, create a page to hold it. You can make the page plain or fancy, whatever you like. You can also name it anything you want (we'd suggest something like "StatusChecker.htm").

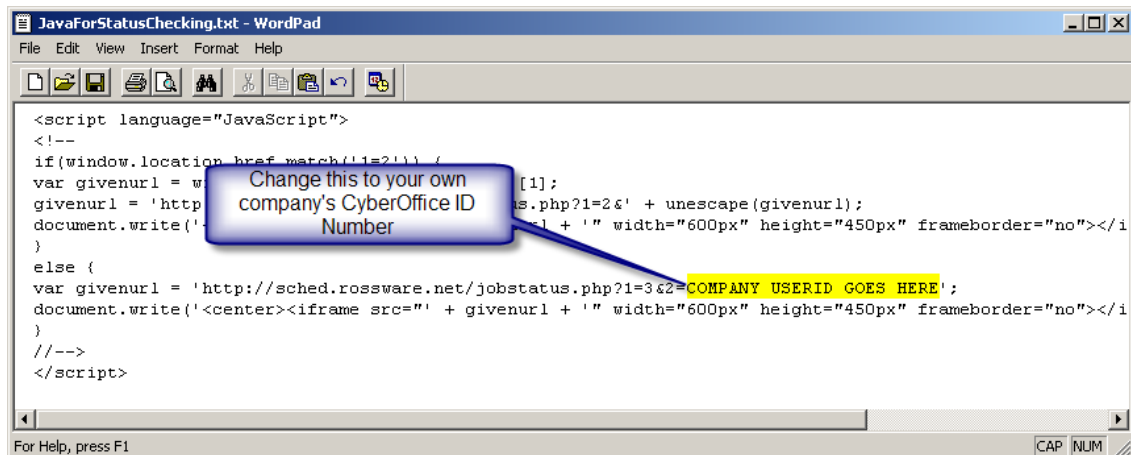
How do you make your status-checking page hold our interface?

It's done by pasting some simple JavaScript (just simple text) into its html code. We provide the text for you. It's contained within a text document called 'JavaForStatusChecking.txt.' You may find this document via the *JavaScripts* link on our CyberOffice downloads page at:

<http://rossware.net/downloads/SDCO/>

There is one little element of editing you must do within the provided text. There's a place where it must have your own company's CyberOffice UserID (a 4-digit number) – in replacement for some dummy text that's otherwise there.

Here is an illustration showing, in general, what the simple text looks like, and where you need to substitute with our own company's ID Number:



```
JavaForStatusChecking.txt - WordPad
File Edit View Insert Format Help
[Icons]
<script language="JavaScript">
<!--
if(window.location.href.match(/1=2/)) {
var givenurl = window.location.href;
givenurl = 'http://www.rosware.net/jobstatus.php?1=2&' + unescape(givenurl);
document.write('<center><iframe src="' + givenurl + '" width="600px" height="450px" frameborder="no"></i
}
else {
var givenurl = 'http://sched.rosware.net/jobstatus.php?1=3&2=COMPANY USERID GOES HERE';
document.write('<center><iframe src="' + givenurl + '" width="600px" height="450px" frameborder="no"></i
}
//-->
</script>
For Help, press F1 [CAP NUM]
```

So, get the text from the above-described source, modify as instructed, and place it within your status-checking page's html. Once you've done that, your status-checking page (as such, and aside from being equipped with data) should be fully operational.

Of course, a page by itself is none too handy if people don't have a way to get to it. Addressing that is our next step.

Just as with Scenario 1 (up-front on-line booking), it's a good idea to have one or more buttons (or menu options) on your site, where people can click to go your status-checking page. Create such buttons where you want them, label as wanted, and simply make them hyperlink to your status-checking page, using its direct url (this is in contrast to Scenario 1, where you must use a url that points directly to a Rossware resource). The JavaScript, as embedded within the page, will take care of everything else.

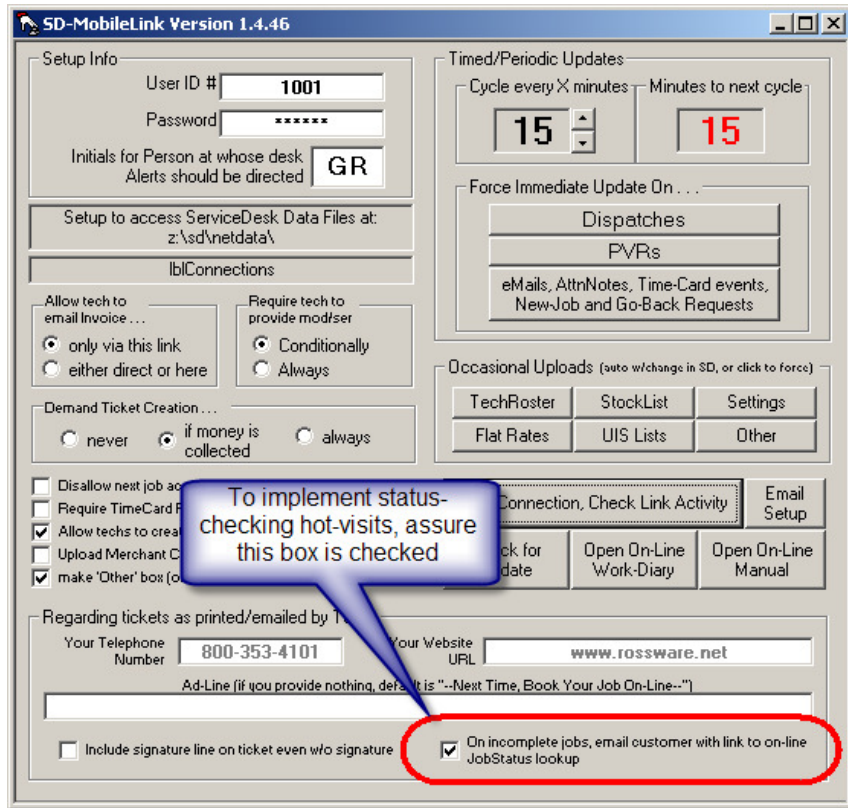
The above-described buttons (or menu options) will nicely accommodate what you might think of as the "*cold-visit*" situation – where your customer has independently browsed to your website and gets to your status-checking page by clicking on your buttons. In that situation, the interface will demand that the customer provide elements of information sufficient to verify who they are, and to identify the particular job on which they are seeking to know the status.

We believe a very important aspect of status-checking is to also accommodate "*hot-visits*." In this situation, by contrast, you'll be emailing your customer with specific notice they can check on their job via your website, and the email will contain a link on which your customer can click to go direct to the status-checker as applicable to their particular job (i.e., without having to enter any information for the purpose).

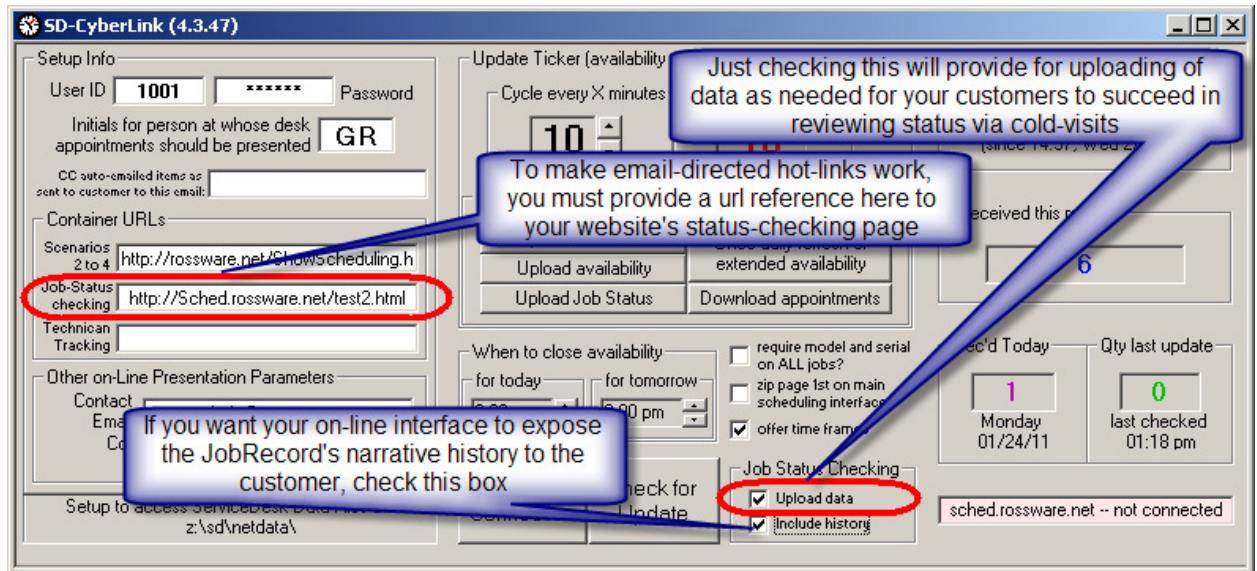
At this time, we have programmed for creation of one such hot-visit, direct-link scenario (more are coming). It's via SD-Mobile. The basic concept is, if your tech does his PVR via Mobile, and if as part of his report indicates the job is not complete, the MobileLink program will automatically email the customer, express regret the job could not be completed, and provide a hyperlink upon which the customer can click to review progress on the job (and that's all it will take, is just that one click).

To make the above feature work, you must pay attention to settings in both the SD-MobileLink and SD-CyberLink programs.

In MobileLink, look for a checkbox at the very bottom-right, and assure it is checked:



In CyberLink, look for the following:



It should be that easy.

As intimated, we're intending to make other auto-email-with-hot-link actions – such as, for example, possibly when the job is first created. Please watch for such developments.

Details for Operating Scenario 6 (Technician Tracking)

Setup for this scenario closely parallels what's done for Scenario 5. In fact, it parallels so closely you may use the same instructions as found in the 2nd through 7th paragraphs in the preceding section, and simply adapt, as obviously applicable, for this scenario.

Where a substantive difference arises between these scenarios is in the method used to provide your customer with an email hyperlink that enables “*hot-visits*” to the particular web page in question (here, it's your tech-tracking page, as opposed to your job-status checking page) — in particular, in providing an email that gives your customer a hyperlink to open the page as *pre-setup* for a particular job, as opposed to merely having “cold-link” buttons provided on your website.

You may recall from the prior section that we are presently configured to have the *SD-MobileLink* program optionally send emails after any visit where the tech did not finish. These emails express regret the job was not completed, and provide a link the customer may click on to monitor the job's progress (i.e., Job-Status Checking).

The parallel for that here is simply in a different place and operation. Specifically, it's provided via the *CyberLink* utility itself (i.e., is not “farmed out” to MobileLink), and, instead of being paired to the tech's non-completion after a visit, it's paired to the Scenario 4 Confirmation process.

Specifically, after ServiceDesk emails the customer a request to confirm a next-day's appointment, and after the customer then proceeds to your website's online interface to make that confirmation, the CyberLink program immediately follows with an email that says, essentially, “*Hey, thanks for the confirmation, and . . . incidentally . . . if tomorrow you happen to be wondering how your tech is running, just click on this link.*”

Presently, the CyberLink program will do this automatically any time it downloads an appointment confirmation — assuming, at least, that you've otherwise set it up to accommodate technician tracking. In other words, we have not presently given you an option to NOT have such emails sent, *if* you're otherwise using the confirmation feature, and *if* you've otherwise setup your system to offer technician tracking. If you happen to *need* an “off” feature for this, you'll have to let us know.

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 a mere 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 'JavaForScenarios2To4.txt' You can download this from CyberOffice downloads page at:

<http://rossware.net/downloads/SDCO/>

3. In CyberLink, there's a section titled "*Frame URLs.*" The first box in that section is labeled "*Scenarios 2-4.*" 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).

- A. 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\netdata folder on your server. There is a single, simple caveat: Likely, you'll want to have your text recite back to the customer details, such as the date the appointment is for (perhaps also the name that it's under and address). If so, the solution is very simple. At the locations in your text that you want any such "fields" to appear, place in field designators as follows:

[AppmntDate]
[Name]
[Address]
[CityState]

Be sure to include the brackets and all. These little specific snippets of text are specifically what CyberLink will look for when configuring your email. It will replace those designators with actual text as applicable to the customer and appointment in question.

B. 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-AfrFrstAppmnt.TXT*. Use the same method as above-described to indicate where, within your text, you want particular "fields" to appear.

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

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

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

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

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

There are actually two choices for types of email that can be sent to remind a customer of his or her appointment, and to request confirmation: (a) one requests confirmation via email or telephone reply (consider it the "old fashioned" method); (b) the alternate method requests confirmation via the customer clicking on a hyperlink, which takes him/her to an appropriate CyberOffice interface on your website, with action there linking automatically back into ServiceDesk.

You may configure custom text for either or both scenarios. Regardless of which, you'll need to follow the same *pattern* of instruction as described in regard to customizing for Scenarios 1 and 2. Specific differences, though, are as follows:

For confirmation Type A (old-fashioned method), the filename for your custom text must be *MyTextForAppmntCnfrmtn_MethodManual.txt*.

For confirmation Type B (modern/CyberOffice based), the filename must be *MyTextForAppmntCnfrmtn_Automated.txt*.

In either case, your custom text may be configured to employ the following fields:

[AppmntDate]
[PosInSequence]
[TechReference]
[EstimatedTimeFrame]
[JobDescription]
[CompanyTelNbr]
[Hyperlink] *(this field used solely in Type B confirmation)*

It's not essential that every field be used (it's not really required to indicate who the tech will be, for example), though some are obviously essential (such as the hyperlink field, for example, if using the Type B confirmation). Aside from essentials, please use as needed to optimize your communication.

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 Rossware 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.