

SD-Mobile Handbook

SD-Mobile is a cutting-edge system for automating information linkage between the field-tech and office. Working entirely behind the scenes, it provides each field tech with comprehensive and real-time data concerning the dispatches assigned to him, including additions and cancellations throughout the day. It simultaneously provides the office with real-time data concerning each tech's progress, including all pertinent details concerning what transpired at each job.

In short, SD-Mobile is a 21st Century system for the truly modern service business.

This handbook assumes you've contacted Rossware to request setup of your Mobile account on their end, and have received your "Setup" email. Congratulations on your choice.

This manual is designed to help you begin operation easily, quickly and with almost zero fuss. It's barely 23 pages (portions of which can be skipped if not applicable), and is easy reading.

Chapter 1

How the System Works

The SD-Mobile system involves two programs:

1. **SD-Mobile** is the program that runs on each tech's computer. It's what he uses as he works in the field. There two versions: one for Windows (full Windows is required, as opposed to Windows-RT), and one for iPads. Either way, the device will need an internet connection. At bare minimum, it must be connected at the beginning of each day. Far better is to be connected at least intermittently throughout each day. The connection is typically achieved via a built-in-to-the-device "air card" (sometimes called "cellular") or via an external/mobile hotspot (such as is provided by newer smartphones or by a MyFi-type device). For "free" access some people use WiFi as found at localized "hot spots."
2. **SD-MobileLink** is a program that installs and runs at the office, from any ServiceDesk-networked computer. It should run on one machine only, but should be *kept* running throughout the day. Its task is to provide the *office-end* of communication with the tech (it's the "messenger," in other words, that shuttles information back and forth, between your techs and ServiceDesk in your office).

The general scheme is as follows:

1. We maintain a remote server that provides a common information point, to facilitate communication between office and tech.

2. SD-MobileLink (running in the office) uploads info to this server, as applicable to the techs your office has in the field (i.e., their names, log-in passwords, etc.) and their work for each day (i.e., jobs scheduled, details concerning those jobs, etc.).
3. SD-Mobile (running on each tech's mobile device) logs into the remote server (using the tech's log-in credentials), and downloads dispatch/job data as applicable to his work.
4. The tech proceeds to each job, as directed by his SD-Mobile interface. As he arrives, there's a button to check-off his arrival. As he finishes, there's a similar button. As he does each, info concerning the fact uploads to the remote server.
5. Meanwhile, at the office, information regarding these arrivals and departures are downloaded by SD-MobileLink, and show graphically within the ServiceDesk DispatchMap. The events are also recorded in each job's narrative history.
6. Upon completing each job, the tech should do his PostVisitReport (PVR), via the provided SD-Mobile interface. This information likewise uploads to the remote server, and downloads (via SD-MobileLink) to the office and ServiceDesk -- in real-time (or at least nearly so, depending on the tech's connection status to the internet).
7. In addition to doing his PVR, the tech may also print (or email) a very nice retail ticket for the customer. This means there is no need for the tech to bring any paperwork with him, and fulfills the vision of a "paperless" office that many have yearned for.
8. Offices *could* opt to do PVRs later (i.e., rather than in real-time by the tech using SD-Mobile), but we don't believe many will. The advantages of the tech doing so directly are too great. Plus, it's by doing the PVR that he assembles the info (at least in part) that goes into printing the customer's ticket.
9. Programming in SD-Mobile is structured so, for any period during which the tech's mobile computer lacks an internet connection, it simply holds the info needing uploaded, and sends it in a burst the next time a connection is available.
10. A further aspect of the real-time communication is, if the office makes changes in the tech's schedule throughout the day (additions, cancellations, etc.) they also show in real-time (or as soon as the next internet connection is established) within SD-Mobile.

For the most part (and with understanding of the general scheme, as explained above), use of the two programs will be self-explanatory.

Chapter 2

Some Details Regarding Setup

To begin your use of this system, you must first run the two installer utilities (one for *MobileLink* in the office, and the other for *Mobile* itself at each portable computer that you're setting up for a tech).

Both installers are found on the SD-Mobile downloads page. To get there, you can use this link:

<http://rossware.net/downloads/SDM/index.html>

You can also navigate there via menu options found on Rossware's main webpage, <http://rossware.net> (also available via hyperlink in the ServiceDesk *About* form). Simply choose *Updates*, then *SD-Mobile*.

To run either installer, first click on the applicable hyperlink, then (when prompted) choose "open" or "run." After several moments, the system will open a folder that contains three files. Click on the file called *Setup.exe*, and follow the prompts. Generally, it's best to accept default answers as the installer setup progresses.

The above assumes your techs will be using the Windows version of the Mobile program (SDM-w). If instead he'll be using the iPad version (SDM-i), it will be downloaded instead from the iTunes store. There is a link to that download from our own Mobile-downloads page (as referenced above), or it can be navigated to by searching directly within the iTunes store (search on "Rossware").

Besides pointing you to the above-described webpage, your initiating email will have also provided you a *UserAccountNumber* and *password*. You'll find both Mobile itself (regardless of which version) and MobileLink have boxes in which these unique credentials must be filled in.

It's also necessary to do some work in *ServiceDesk*.

Specifically, you must go to the ServiceDesk *Settings* form, and there designate each tech that you wish to have running with SD-Mobile (if you do not, his name will not be uploaded by SD-MobileLink, as a tech who's authorized to use the system).

To designate the applicable techs, locate the *List of Technicians* section in your ServiceDesk *Settings* form, and select an applicable tech. Then, in the grey *Properties* box to the right, put a check in the box that indicates he's '*Using SD-Mobile*'. At the same time, be sure to create a password for his log-in.

With this done, SD-MobileLink will know to upload the applicable techs' names (and log-in passwords) to the remote server. This, in turn, allows those techs to log-in via SD-Mobile.

In this regard (and when you look at the SD-MobileLink program), you'll see it's structured to update several different kinds of data, including your TechRoster. It's set to do this in timed intervals. In regard to updating Dispatch info and PVRs, the system connects to the remote server with each timed event regardless of whether changes have been detected in ServiceDesk. In regard to uploading the TechRoster and several other matters, however, it only uploads when it detects changes have been made.

Regardless of the above, let's say that within ServiceDesk (and as per above description) you just designated some techs as '*Using SD-Mobile*' and do not want to wait for the timed update event for SD-MobileLink to upload this fact to the remote server. Within the SD-MobileLink program, you'll see there's a button you can click to make it happen immediately.

There are similar buttons to force immediate uploading of your *StockList* (used by SD-Mobile for the tech to indicate parts used from stock), *UIS Lists* (provides your list of Types, Makes and Selling Dealers so the Mobile tech can have drop-downs for same), and so on.

Chapter 3

Hardware Considerations

For use in the office, no new hardware is needed. You already have one or more office computers, and presumably have a good internet connection there as well.

A. The Mobile Computer

For the mobile end (and as previously mentioned), SD-Mobile may be used in either a full-Windows platform or in an iPad.

If preferring a Windows platform, netbooks were a once popular solution (essentially they were very small laptops with limited computing power). More recently, netbooks have been superseded by Windows tablets, so, pretty much, your Windows-side options are between a laptop versus tablet. Regardless, bear in mind that tablets can indeed easily be configured in the traditional clamshell-with-keyboard mode, either via inherent design (e.g., the Asus Transformer or Microsoft Surface, etc.) or via the add-on of an accessory sleeve/keyboard. Historically, the availability of incorporated air-card/cellular capability lagged, in the Windows-tablet world, but as of early 2015 this is rapidly changing.

If preferring the Apple/iPad path, your options are obviously much more simple, because there is only one line of devices from which to choose. Most likely, you will want to add an accessory sleeve/keyboard. Most techs find the iPad Mini is a very convenient size. Most companies are choosing the upper-end model that includes built-in air-card/cellular capability. Any iPad that's running iOS 6 or above should be fully compatible.

Regardless of whether choosing the Windows or iPad route, for at least the purpose of running SD-Mobile, you do not need extra computing power and/or extra ram. The base-level devices will perfectly suffice.

To give you some idea on pricing, a bargain-basement Windows tablet plus add-on sleeve/keyboard can be obtained for as little as \$200. That's pretty cheap. Alternatively, you can get something like the Asus Transformer (a really nice device) for about \$350. Just recently, we saw an aircard/cellular equipped HP tablet offered at just \$149 (from Walmart; it does not include a keyboard/sleeve), and the aircard may be configured to work with any of the major carriers.¹ If purchasing a Windows device, please be sure it is *full and true Windows* (i.e., not *Windows-RT*, which is a much more limited operating system that only masquerades under the "Windows" name).

Apple devices generally cost more. On checking presently, the cheapest is \$249 (stand-alone, no sleeve/keyboard), and lacks built-in air-card/cellular capability. The cheapest with that capability is \$429. Regardless, the feedback that we've received from those companies that have deployed both is that their techs vastly prefer the iPad solution.

As for internet connection (and as mentioned), you can do it internally within each device via built-in capability, or externally (and the choice is entirely up to you). Aside from the connecting hardware, you will also need a data plan. We suggest you internet shop, to see what options are currently available to you, and decide what best fits your need. As for data amount to expect SD-Mobile to use, our reports from users are that most techs fit well and easily at under 2 GB per month.

B. The Mobile Printer

In the early days of SD-Mobile, quite a few of our users wanted to print tickets for their customers. Today, very few have this preference. Regardless, SDM-w possesses ticket-printing ability in spades. Being newer (and having been created in an era when few wanted such printing), SDM-i does not have such ability.

If you are using SDM-w and wish to print, this section is for you (otherwise, we recommend skipping over it).

As for which printer to use, it's entirely up to you (SDM-w will auto-adapt its output for whatever paper size your device uses).

Some guys have smallish (albeit still office-size) inkjets mounted in the truck. A virtue is they're cheap. A downside is the tech has to go out to the truck to get the customer's ticket.²

¹ This was March of 2015.

² We don't know of anyone doing this yet, but an idea we've had is you could put a wireless hub in your truck, power both it and a standard/cheap (but networkable) office printer via an inexpensive inverter, connect your printer to the hub, and use a WiFi enabled laptop (most are, these days). You remain in the house while tallying the ticket, click on Print, and the ticket begins printing in the truck before you even walk out there. You then go out to your truck with your tools, put them away, clean up, and go back into the house with a perfectly completed ticket.

There are a number of more compact, “mobile” inkjets on the market, and if you check the pricing you’ll see there’s a significant premium for going tiny. For years, we’ve been happy with a Canon ip90. It’s pretty small, yet prints to full, letter-size paper (we have a client who keeps his ip90, along with other supplies, in briefcase that he carries in with him). Figure around \$240 to purchase.

Another option is the little micro thermal printers that store their paper on a roll. We acquired a designed-for-service (i.e., “ruggedized”) Printek RT43 BT that connects via Blue Tooth and also includes a credit card reader (and may optionally be attached to a barcode reader). It’s a very sweet machine (can be worn on a tech’s belt), though some may consider pricey at around \$750 typical retail. We recently became a Printek dealer, and with that status can provide you with significant discounts from list (price sheet at <http://rossware.net/MiniManuals/Printek-PriceList.pdf>).

For the purpose of our tradeshow demo-ing, we also purchased the very-micro *Brother MPrint 140BT*. It has the size and approximate appearance of an old-fashioned cigarette case. It prints individual 3" by 4" sheets, and also connects to the computer via BlueTooth (i.e., no wires). Ran us about \$370. Very cool.

C. The Credit Card Swiper

Just as there is nothing in SD-Mobile to mandate mobile printing, there is likewise nothing to mandate live and in-the-field processing of credit card transactions. Nor is there anything—even if you want your techs to run such transactions live—to mandate their use of a swiper (i.e., our Virtual Terminal happily permits *keying in* of credit card data).

However, given the fact that many merchant-account setups penalize non-swiped transactions, and of course it takes less time and effort to swipe, we think you’ll likely want to make your techs swiper-equipped.

We also think you’ll almost certainly want to have them do that swiping in a manner that’s fully and direct-integrated with SD-Mobile. Any other mode is going to cost you more (in SD-Mobile, there is no fee aside from the standard Merchant discounts). Plus, any non-integrated system will involve separate entry of each transactional detail, thereby involving more tech time and greater possibility of error (as opposed to the perfect integration that’s involved when running within a unified and coherent system).

In fact, with the SD-Mobile unified credit-card-charge system, all you need for swiping is a simple MCR (magnetic card reader) device that can be had for less than \$50 each. On the Windows side, you can use any Magtek HID-type USB-connecting reader (we specifically recommend their model 21040140). On the iPad side, please use the IDTech Shuttle.

For other details on using our built-inVirtual Terminal system, please consult this document:

<http://rossware.net/MiniManuals/VirtualTerminalHandbook.pdf>

D. The BarCode Scanner

In terms of barcode reading, SD-Mobile has presently been configured to work, specifically, with the *Printek* line of devices, as described two sections back (in the section on printers).

In fact, SD-Mobile will certainly work with other devices—any that you can “turn-on” via a provided Windows interface, and whose textual output can be directed to the particular textbox in which you happen to have placed your Windows focus (attributes that should apply to most general-purpose barcode reading devices).

The Printek printer/scanner is a special case, since the printer/body/brain requires a particular command from the computer to activate its scanner, and Printek does not provide a generic Windows interface for this purpose. For that reason, at the bottom of SD-Mobile’s “PVR” page, there’s a button for the purpose.³ If you are using a different scanner, you will not need to worry about this button.

In general, we do not think there is a major need, in at least most operations, for techs to be equipped with a barcode scanner. However, at bare minimum scanners offer a *potential* convenience, and in some situations may pay for themselves handily. Essentially, they allow a technician to use the scanner as a means—much quicker and more accurate than typing—of inputting applicable part number, model and serial numbers wherever the same are provided in barcode format.

In regard to having such barcodes available, as January ’09 ServiceDesk is configured to print parts labels, with barcodes, in a much more convenient and easy fashion than ever before.

E. A Possible “Dream Setup”

Many service companies are buying, essentially, two cell-system accounts for each tech: one for the tech’s voice communication (aka cell phone), and another for his data communication (aka air card). It’s a significant expense. Wouldn’t it be nice (pardon the cliché) to “kill two birds with one stone”?

It turns out, there are a couple of ways you can do it.

One Cell-Account Method 1:

Some of today’s cell phones are able to work as a wireless router. In other words, they connect themselves to the cell system (including voice and/or data services), and in turn can provide the connection wirelessly (typically via Bluetooth) to other devices in range—much like a wireless router in your office can provide in-range devices with a connection to its data source.

³ For a simple page of instructions on setting this up, please click on this reference:
<http://rossware.net/MiniManuals/UsingThePrintekBarCodeScanner.pdf>.

We have not canvassed the market to determine which particular cell phones can do this, but we are told that phones running on *Windows Mobile* definitely can (requires addition of a simple little “router” program that you can download off the internet for free).

So, the idea is you have your tech equipped with such a phone, the account that’s it’s on includes data service, and the tech’s laptop gets its internet connection via the phone, likely connecting to it via Bluetooth. One account, and essentially no extra hardware.

One Cell-Account Method 2:

Ever heard of a “Travel Router?” Until recently, we had not. We discovered their existence while seeking a method, at tradeshows, with which to simultaneously connect multiple computers to the internet—while nevertheless paying for but one air-card connection. Turns out you can take your air-card device and plug it into one of these. They work very much like a router in your office—except, while the one in your office connects (usually) to a land-based internet connection, a travel-router connects to the internet via an air card.

In short, a travel-router (CradlePoint is the brand we purchased) provides you with your own personal and traveling WiFi hotspot—based on your purchase of (and subscription to) a single air-card service.

And, with a travel router, you can do the reverse (at least virtually) of what’s described in Method 1. There, you use a cell phone for cell phone purposes, and secondarily as the internet connection tool and router. Here, you use an air card equipped router as your internet connection tool, while secondarily it can provide you with *virtual* cell phone service.

Specifically, we’re hoping you’ve heard of Skype, which is a method of engaging in voice communication via an internet connection. It’s similar to VOIP (voice over internet protocol), but has some advantages. If you have your own WiFi hotspot traveling with you, there’s no reason you could not equip yourself with a Skype cordless phone, get a “to-from” Skype subscription for about five bucks a month, and scrap any separate cell service.

Whether you’re using Method 1 or 2, another potential benefit is networking with other devices (such as a printer) wirelessly. In such regard (and as you weigh pros and cons), bear in mind that a travel router uses the common WiFi protocol, and will have a minimum range approaching 100 feet, with potential distance much greater (depending on model chosen, attached antennae, etc.). This could make it very practical to keep a WiFi-enabled printer in your truck, issue printing commands from inside the customer’s home via laptop there, and have them immediately begin printing in the truck. Though we’ve not studied the market, we think it’s more likely a cell phone acting-as-router will be using Bluetooth, which is much shorter in range.

Another consideration is that virtually all modern laptops have WiFi built-in. Most do not have Bluetooth, though it can be added easily with a tiny USB adapter (hardly protrudes from the slot) at minimal cost.

Chapter 4

About the Software Environment

Besides needing a good swimming pool in which to do its swimming (please excuse this little metaphor regarding the need for a good hardware setup, as just discussed), SD-Mobile also needs fellow swimmers with whom it can get along. In particular, it's important that its swimming pool not be infested with nasty sharks — especially not the Great White Shark or its close cousin. These sharks go by the common names *Norton* (Symantec) and *McAfee*.

Those two anti-virus protection systems are infamous for wreaking havoc with programs like SD-Mobile. If you think about it, it's not terribly surprising. After all, SD-Mobile is engaged in near constant communication with other systems via the internet. To a virus protection software, this kind of activity can (potentially) make it *look* much like a virus.

Of course, a really good anti-virus protection system nevertheless manages to distinguish between a good program (like SD-Mobile) versus a lousy virus. We have found the antivirus protection systems that are included in Windows, and are also free (specifically, it's Microsoft Security Essentials if you're in Windows 7, or Microsoft Defender if you're in Windows 8) generally do well in this regard. And there may be others that also do well. Regardless, Norton and McAfee often fail. And, when they fail, they end up curtailing essential SD-Mobile operations.

For this reason, we *very strongly recommend* that you carefully assure your computers do not have either Norton or McAfee running on them. This is not always easy, because either Norton or McAfee are almost always pre-installed on new computers, and they can sometimes be as difficult to remove as is a virus itself. Please, whatever it takes, do it, and assure you instead have a less troublesome antivirus system protecting you.

Will SD-Mobile succeed in spite of Norton or McAfee being present?

The answer is: "For a limited time, perhaps yes." To be candid, it's hit or miss. Just because there are sharks in the pool, it doesn't mean they'll always (or immediately) eat you. The longer you swim with them, however, the more likely it becomes. Eventually, if you run with either program, you'll likely be bit. Please, for your sake and ours, assure they are removed.

If you do not, the day will almost certainly come when you encounter great frustration because things do not work as per design, and then (wondering what could possibly be wrong) you'll contact us for assistance. We'll then likewise encounter frustration and loss of time — because you allowed those nasty sharks to remain in your pool!

Do not do it!

We all have much better ends to pursue with our time — as compared to bleeding it away via injuries as needlessly caused by nasty sharks.

Chapter 5

A Few Notes on the Mobile End

As previously mentioned, actual operation of the two programs is *mostly* self-explanatory. But a few things on the mobile end may not be obvious. We'll endeavor to here describe those.

1. On the *Job List* page, there's a column labeled "**PVs.**" It stands for *Prior Visits*. If any appointment is other than first, the quantity of prior visits will display in this box.
2. On the same *Job List* page, the very next column is labeled "**Prts?**" This stands for *Parts*. The purpose is to inform the tech if there are any parts (and quantity if so) he should be taking with him (i.e., either as prior-ordered for the job or as speculatively transferred from office inventory). If there are such items and he clicks in the box, the system will display a list showing each item he should be taking (and location where he should acquire it).
3. In connection with any job selected, the *PVR* and *Print* page are not enabled until the tech has clicked on the "*I've Arrived*" button in the *Job Details* page.
4. Within moments of *first-time* connecting, SD-Mobile will solicit permission to download SmartParts data.⁴ At this time, there is no charge for use of such data in the mobile environment. However, the download can be lengthy. Plan to consent only when you expect the mobile computer to be running and internet-connected for a significant period of time. After the download has successfully completed, the data will be stored on the local drive, and need not be downloaded again, at least pending future updates.
5. The *PVR* and *Print* pages have list sections in which line-by-line items may be inserted by the tech (parts used from stock and parts to be ordered). We've equipped these boxes with the standard ServiceDesk convention for *deleting* an entire line item: *right-click* to delete.
6. When the system detects that it's dealing with an OEM warranty job, it will require that the tech provide model, serial, purchase-date and selling-dealer. If the tech is doing an in-stock repair, he can satisfy the purchase-date field by typing "DEALER STOCK".

⁴ Our apologies to those of you in any non-appliance field. We hope to be getting similar data for other fields soon.

7. The *List of Labor Items* (on the *Print* page) serves a dual purpose. Besides being used as a mechanism to list labor items for inclusion as charges in the printed ticket (preferably via drop-down insertion as connected with your Flat-Rate list),⁵ it may also be used as the basis of inserting text to the *PVR* page's *Description of Work Performed* box.

To further explain, let's look at it from the opposite perspective. On the *PVR* page, the tech wants to describe what he did, but doesn't want to expend effort typing out all the words. Instead, he can go to the *Print* page, and (via its *List of Labor Items* box) simply pick items out of your Flat-Rate list.**

When he picks such items via a mouse *right-click* (rather than standard *left*), it signifies he wants the description-portion of the flat-rate entry inserted not only to his listing in the *Print* page, but also to his *Description of Work Performed* in the *PVR* page. Thus, by this means he can populate that latter page with a nice, verbal description—while never typing a single character.

The system is also configured so that, if the tech has not deliberately inserted from the *List of Labor Items* box to the *Description of Work Performed* (per the above), and if he's otherwise left the latter empty, it will query him, when he goes to print, as to whether he'd like the insertion done for him.

Overall, you're going to find that with all systems properly implemented, the tech will be able to do virtually all his work with very close to zero typing (typing in model and serial numbers, when needed and if not previously provided, is one exception).

The vision of a paperless system, with each tech managed superbly and remotely much like the UPS man with his electronic clipboard is here.

Chapter 6

Composing Mobile Tickets

As intimated when discussing hardware, SD-Mobile is endowed with all essential equipment for a technician to formulate a ticket in the field, collect a signature, and provide a copy to the customer. All such work is done (as you might guess) via the 'Print' page.

The general concept on this page is that it has two list areas: one for parts items used, and one for labor items. The first will auto-fill based on parts that were indicated (as being used and/or ordered) back on the *PVR* page. It can then be edited to fit needed circumstances. The second can be filled in manually, or with semi-automation based on a flat-rate system and drop-downs (see applicable section following). It also has an area where charges are totaled, and tax added.

⁵ See Chapter 7.

On the basis of this data, the system stands ready (when needed) to create an actual ticket image.

The first context in which such creation is likely to be wanted is for the purpose of collecting the customer's electronic signature. The simple concept is, the tech clicks on the button labeled '*Review and Sign*,' at which point the system creates the electronic ticket image, and presents it on-screen (in an interface which allows the customer to review and sign; please see Section C following).

The second context is for providing a copy to the customer, which may be done either by printing or emailing.

Regardless of context for which it is formulated, the electronic ticket image is "hard wired" (at least to a significant extent) into the design of Mobile (it's just not practical in this context to reproduce the great flexibility in ticket design that's otherwise enjoyed in ServiceDesk). But some flexibility is allowed (see Section B following).

A. Using Flat-Rates

As mentioned, the Print page is configured to work most easily (in terms of allowing the tech to fill-in a description of what he did and what the charges are) on the basis of a flat-rate system. It's not essential (SD-Mobile will work fine without it), and you should not delay implementation while waiting to get one in place.

However, there are two factors we think should make you serious about setting one up as soon as possible.

One is the difficulty of typing descriptions of work performed in the mobile environment, whether owing to the fact techs are lousy at typing or because of limited physical circumstances (e.g., tiny or no keyboard). Another is the need to quickly populate a ticket, to be printed in the field, with descriptions of work performed, including appropriate charges.

Given these needs, we've designed the system to make very effective use of such a list. But you must provide the particular list you want to use. And, your list must be formatted to the particular structure the system is designed to accommodate. We have a separate document, designed to walk you through setting up your flat-rate list (in fact, we even provide a beginning list for you). Here is the link to that document:

<http://rossware.net/MiniManuals/FlatRateSetup.pdf>

B. Print vs. Email

Obviously, a tech may provide a ticket to the customer via printing, and a nice '*Print*' button is provided on the *Print* page for this precise purpose. However, it's not the actual 'Print' button that does the work; it's a physical printer. Pros and cons of various printer strategies were discussed in the section on hardware. A commonality you might have noticed is that all involve some level of burden and expense.

We live in an increasingly electronic age, and here at Rossware we think it makes a lot of sense to avoid the printing burden entirely. More and more consumers are growing used to (and comfortable with) receiving tickets electronically, via email. Why not take advantage of that?

It's true that some customers will either be unable or unwilling to accept their ticket via email. However, this should not in itself be the basis to decide against such an option. For whatever percentage of customers as need a paper tickets (most servicers are finding that percentage is surprisingly small), the tech can simply maintain a supply of blank paper tickets, and hand-write the essentials as needed by the customer.

An emailed ticket will look exactly like a printed one (except it has a yellow background). It's done in the common .jpg format, so any customer with a computer should have no trouble opening it. An added benefit in using this method is it gives you an excuse to collect the customer's email address. There are just two issues to keep in mind:

(1) The new ability is configured so that, at his option, the tech may either email the ticket directly from his mobile computer, or he can click a button requesting that SD-MobileLink (back in the office) send the item from the computer in which it's running. I suspect you'll prefer the latter option. Disadvantages of the first include the fact that, if sent from the tech's mobile computer, the customer would get access to his direct email address. Plus, he'd need to have Outlook Express correctly configured in his computer (to do the actual sending). If all sending is done via the office computer, both worries are ameliorated.⁶

(2) At any machine that's going to email invoices, an underlying file is needed that was not part of installations prior to Ver. 1.0.36 (it's what allows the system to formulate an image in .jpg format). The file is called *FreedImage.DLL*, and it's available on the SD-Mobile downloads page.

Whether the ticket is emailed or paper-printed by the tech, the system will create an electronic image of it. MobileLink will save this image to the ServiceDesk *HLinks* folder, and place a link thereto in the narrative history as added to the applicable JobRecord (i.e., as part of its PVR entry). At least, the above occurs *if* the system is equipped with that *FreedImage.DLL* file.

C. Customizing Your Mobile Ticket

The standard mobile ticket contains a few items of text that will easily adapt to your circumstance, specifically in the header area at top. In particular, you can (and should) specify the telephone number and website url you wish to have appear on the mobile ticket. Just put these in applicable boxes of the SD-MobileLink interface, and the rest will take care of itself. Also, if you want to use a different ad-line than the standard/default one ("*-- next time, book your job on-line --*"), just type it into SD-MobileLink's applicable box.

⁶ You'll find in the MobileLink program there is an option to *disallow* the tech from direct-sending tickets (thereby forcing him to do it via the office). I suspect most of you will want to choose this option.

If you'd like your tickets to look a little more stylized (the above-described header is all just plain text), you may replace the entire standardized header with an artistic/graphic one (i.e., containing your company's stylized logo, and/or other accouterments). To do this, simply create the graphic image you want, and save it to your server's `lsd\netdata` folder under the name `SdmLogo.jpg`.

A final element of potential customization⁷ involves the signature line. Absent you doing anything otherwise, the text immediately under it will default to something like: "*I promise if I don't pay this bill you'll own my house, my cars and all my children.*" (okay, a little less extreme). If you'd like different language, open any text editing program (NotePad, WordPad, Word, etc.), and type the text you want. Then save. Make sure to save it as a "Plain" or "Text Only" document. Save it to the `lsd\netdata` folder on your server, as filename `SdMobileInvoiceText.TXT`. Once you've done so, SD-MobileLink will see the file, upload its text, and each of your Tech's instances of SD-Mobile will download the same, and use it in lieu of the written-by-us canned text.

Chapter 7

Electronic Signature Capture

Beginning with SD-Mobile Version 1.1.0, we added the ability to electronically capture a customer's signature.

The simple idea (as reviewed in the prior chapter) is SD-Mobile displays an image of the ticket as it's about to be printed or emailed (there's a *'Capture Signature'* button on the Print page for this purpose).

He'll likely say something to the customer like "Would you please review this ticket and sign it."

When the customer is ready to sign, she clicks the *"Sign"* button (or the tech can just hit 'Enter' on his keyboard). This visibly enlarges the ticket image until its signing space

⁷ We are occasionally asked for more extensive customizability of the Mobile ticket, such as is offered within ServiceDesk itself. Presently, this is not offered. Nor is Mobile equipped to create the same kind of tickets as offered within SD. There are reasons.

It requires a large programming infrastructure to accommodate layout-customizability in a ticket format that is simultaneously auto-fillable (with particular job-data in each and every instance), user-editable (with edits saved, potentially even in multiples), and printable, via a customized-layout-adapting on-screen interface. To provide some indication of how difficult this is, we don't believe any of our competitors has ever done it. We've done it within SD, but with huge investment, and in a context where — given the relatively large in-office platforms involved significant screen space and direct, in-office management as needed for setup and maintenance — accommodating such infrastructure is relatively feasible.

All such factors are lacking in the Mobile environment, and there's the added factor that (for technicians and in platforms that are field-deployed) there's a critical need to keep setup and deployment simple and trouble-free. Given such factors, we judged it best in this context to use a universal ticket format, maximally adaptable to every context and need (even to printing on tape-type printers, if that is the type employed by a given user).

There is, of course, a downside.

For every gain in design, there's a cost. In gaining simplicity and robust freedom from complication, we lose the customizability you might otherwise prefer. There is reason for the expression: You can't have your cake and eat it too. The world is full of compromises. We have aimed for the optimum compromise in this context. Your opinion may vary.

None of this means we'll not add to user-selectable options, much as has been the pattern to date. It's further possible (to the extent demand is apparent) we'll offer dramatic variations. Your input in such regard is welcome, but as you provide it we urge you to be mindful of the dynamic above-described, which urges so strongly for simplicity. That dynamic will not go away.

occupies most of the screen. At this point, *any pointing device* may be used to scribble a signature.

Here's a list of some possible pointing devices:

1. The mobile computer's touchpad. Assuming you've not invested in a tablet-type computer (where a stylus or finger may be used directly on-screen), this is the most simple and direct method. It can be done via direct finger method, though signing via finger is a little awkward (perfectly possible and doable, but awkward).⁸ Far better is to use a capacitive-type stylus. These have been available in the last few years as specifically developed for iPads, but work on laptop/netbook touchpads as well. Just Google "*ipad stylus*", and you'll find hundreds of options, and at near-token prices.⁹
2. An ordinary computer mouse.
3. A stylus on the computer screen itself (if your tech is using a tablet-type PC).
4. A pen device. This is a small accessory that contains a little pad area for working with a stylus. There are a wide range of such devices, ranging from those designed for serious computer art, to those more targeted to kids (yes, so that a child can scribble on a pad and see the result on-screen). For development here, we acquired a Genius G-Pen 340 for about \$35. It's quite small, cheap, and works superbly.¹⁰

Regardless of pointing device used, once the signature has been scribbled, the customer clicks the "*Accept*" button (or, again, the tech can just hit 'Enter'). This saves the signature image with the ticket. Thus saved, it will print with the ticket (if printing is done), or be included as an emailed ticket image (if that's the method), and of course will be saved within ServiceDesk data back at the office.

Chapter 8

Using Special Disclaimers

We just discussed electronic signature capture as connected with tickets (aka "invoices"). There is also ability to similarly capture electronic signatures, but in

⁸ When first contemplating this method, it was our estimation that to ask a customer to sign with her finger would be somewhat tacky. However, upon doing pretend scenarios here in our office (and with real equipment), we've come to feel it's a rather viable option. A clever technician could make it interesting, novel, even fun: "Hey, I bet you've never signed with your finger before . . . this is new technology," or "How long since you finger painted?" The point is, a little cleverness or levity could make it just fine.

⁹ With touchpad or mouse, it's necessary to hold down the mouse button (or its equivalent) while signing (essentially, ink does not flow from the pretend pen unless the button is down). If your tech wanted to make it easier for a tech-challenged customer (we're thinking 96-year-old Mrs. Ambercrombie), a good trick would be to keep a small wireless mouse in his pocket. He could simply reach into his pocket while she's signing, and hold down its button. Thus, she'd be using stylus on the touchpad, and have no need to simultaneously do anything else.

¹⁰ Purpose-made signature capture devices tend to be much more expensive (a few hundred dollars), and in our experience are not generally configured to mimic mouse input. Unless one can mimic mouse-input, it will not work with Mobile. In general, we are not recommending purpose-made signature capture devices.

connection with *special disclaimers* (e.g., the customer agrees to hold you free from liability if the tech damages her floor while moving machinery).

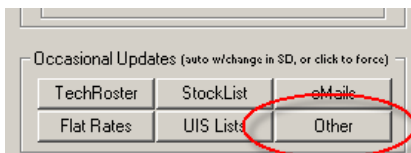
Usage is very simple.

First, you must provide one or more disclaimers. By present design, the system allows you up to six (if more are needed, let us know).

If you wish to “from-scratch” create a disclaimer, open any text editor (e.g., Word, WordPad, NotePad, etc.). Type the desired *Header* (i.e., the “Title Bar” you want to have appear at the top of your disclaimer), then a vertical bar symbol (“|”), then the text *Body* you want to use.¹¹ Once all is typed and perfected, save it as a *plain text* file. It must be saved to the `lsd\netdata` folder on your server. If it’s your first and/or only disclaimer, name it *SdMobileDisclaimer1.TXT*. If it’s the second, name it *SdMobileDisclaimer2.TXT*. And so on. (The naming must be precise, because it’s only for those particular file names, and in the folder location described, that *SD-MobileLink* will look for the applicable purpose.)

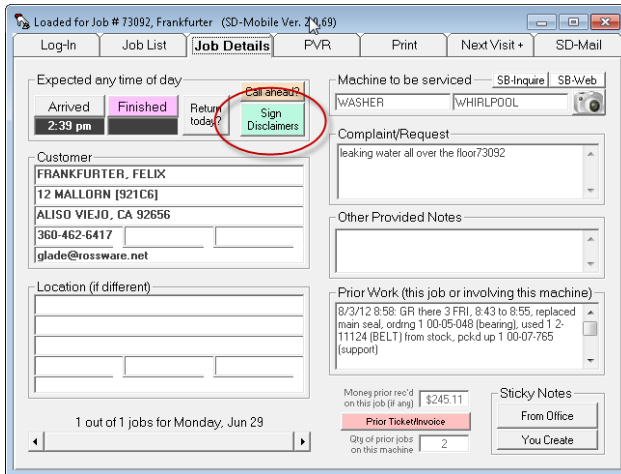
Much as in the case of your customized ticket/signature language, once you’ve done the above, *SD-MobileLink* will see the file (or files), upload each, and each of your Tech’s instances of *SD-Mobile* will download the same, and offer it (or them) for use in collecting signed disclaimers from customers in each applicable instance.

If, incidentally, you want to assure your creation/editing work at the office is immediately uploaded to the remote server (i.e. rather than waiting for the next timed event), you can click on this button in *MobileLink*:



To find, from the *Mobile* side, where to invoke the disclaimer option, just go to any job’s “*Job Details*” page. You’ll see a button as follows:

¹¹ You may elaborate further on this if desired. In particular, if you want any particular disclaimer to be mandatory (i.e., your techs are *required* to have the customer sign this disclaimer), you may add text after a second bar symbol (i.e., in a third section of text, where each section is separated by a bar symbol). This section of text may contain the word “ALL” if you wish for the disclaimer to be required on all jobs, or — if you want it only to be required when particular kinds of machines are being serviced — you may list those machine descriptions, separating each by a comma (or, if only one machine type is involved, simply list it alone with no commas). If listing machine types, be certain the spelling is exactly the same as you created in the ServiceDesk machine-types dropdown.



If, rather than composing your own disclaimers from scratch, you'd prefer to at least *start* with a very nice disclaimer set that has already been created, we have in fact provided a very nice set for you. You can open and/or download this set at:

<http://rossware.net/MiniManuals/Disclaimers.zip>.

In the alternative, you can just go to the *SD-Mobile Downloads* page on our website, and find a link there.

If you are sufficiently fond of our example provisions, you can use them precisely as provided—with one exception. You must *change the company name* (as embodied within the text of each) to match your own.

At the least, our set of examples should provide you with ideas to use in creation of your own. In actual fact, however (and assuming you are in any service business that makes their particular subject matters applicable), we recommend that you use them with very minimal editing, aside from company name. They've been carefully composed with a good understanding (based on solid legal credentials and experience) of what most persuades most judges to take a disclaimer seriously. To be candid, that is a tall order. As a rule, most judges will mostly ignore most disclaimers. To have any chance of being taken with seriousness at all, a disclaimer generally needs the very qualities that have been built into these examples. If you change significantly from what is provided (and aside from incidentals), there is a very good chance you will be creating a weaker (rather than a stronger) result.

BTW, in any instance where a disclaimer is signed, the *Mobile/MobileLink* system will transmit it back to the office as part of the underlying PVR information. In result, you'll see a reference to each signed disclaimer in any applicable JobHistory. That reference will include a hyperlink. To actually view a signed disclaimer, just double-click on the hyperlink.

ne | 2/1/09 10:42: GR there 2/1 SUN, 8:46 to 8:46, estimate approved,
tion | order parts [AC1004], prntd tckt [SdLink\T57.jpg], cstmr sgnd
 | Dsclmr [SdLink\D15.jpg], cstmr sgnd Dsclmr [SdLink\D16.jpg],
 | cstmr sgnd Dsclmr [SdLink\D17.jpg] via SDM)
in |

In the above example, we signed three different disclaimers in *SD-Mobile*, and uploaded the PVR. Hence, there are three resulting disclaimer hyperlinks in the resulting JobHistory (there's also a hyperlink for the underlying ticket, which is not circled above).

Chapter 9

Remote Booking of Return Appointments

One of our very first enhancements to the system, above base-level operation, was to add (on the mobile-end) a “Scheduling” page. This is a locus where the technician, supposing he's ordering parts and is expecting to return, can pre-book his return appointment.

Those companies that have been using this practice (i.e, not within *SD-Mobile*, but via other means) swear by it. They say it makes the customer happier, and saves the office significant resources that are otherwise needed for *it* to re-schedule. The practice presupposes that parts will arrive within some normally expected period, but evidently that factor has become sufficiently reliable that the occasional need to cancel, because expected parts did not arrive, becomes the exception rather than rule.

SD-Mobile's Scheduling page presents the tech with a calendar. The intent is to allow the tech to schedule *intelligently*, by informing him (via the calendar) of the particular days in which there are still openings available for the return visit (taking into account the customer's zipcode and the JobCount value that's expected for the return visit). Specifically, the calendar will show in **bold** those dates that are so available.

At least this will happen *if* *SD-Mobile* has access to appropriate availability data. This, naturally, raises the question: where does it seek such data? The simple answer is it looks on-line for *SD-CyberOffice* data as applicable to your company.

If you do not know, *SD-CyberOffice* is the system that, among other things, allows your customers to book their jobs on-line. To make *that* system work, *it* had to have a means of keeping your on-line interface informed of availability. Thus, we already created such machinery for *SD-CyberOffice*, and could see no sense in re-inventing it for *SD-Mobile*. Instead, we've designed *SD-Mobile's* re-booking apparatus to “piggy-back” on the same machinery.

For this reason, you need to implement at least the basic tools of *SD-CyberOffice*, if your tech is to re-schedule *intelligently* via *SD-Mobile* (otherwise, he can still use the scheduling function, though it will be without knowledge of availability).

If you're already using SD-CyberOffice, use of its data by SD-Mobile is absolutely automatic. Otherwise (and assuming you want your tech to be able to schedule *intelligently*), give us a call.

Chapter 10

Using the TimeClock Feature

Almost as soon as we rolled out the first version of SD-Mobile, some folks wanted provision for their tech's to "punch the TimeClock," via it's interface. So, we added the functionality.

Basically, the SD-Mobile interface has a button via which the tech can punch in, and another with which he can punch out. Based on such actions, the MobileLink program creates a file, for each such tech (i.e., any that uses the feature), that tracks such punch-in and punch-out events. Look for such files in the `lsd\netdata` folder on your server. For any tech, look for a file called *TimeClock.XX.TXT* (where *XX* is the two-letter abbreviation for the tech wanted).

I suggest you open the file in Excel, as that will nicely separate the columns, and provide a context for easily adding times for the period of interest.

You'll notice two columns that may provide a bit of curiosity: they're labeled "InFudged?" and "OutFudged?". This indicates, simply, if the technician entered an In or Out time manually. The system allows him to do this, based on the fact he may have forgotten to log in, then find he needs to log out, or vice versa. However (and as you'll see) it flags any item so entered.

You'll notice that at present the data in these files does not integrate with the ServiceDesk *WageReport* feature. The reason is, over the years we found certain vulnerabilities in the way we'd designed the data as connected with ServiceDesk's *internal* TimeClock function. We designed the new structure to be more robust. Eventually, we need to convert ServiceDesk itself over to use the new structure. Once we do that, the internal-to-ServiceDesk WageReport will work with Mobile-generated data.

Chapter 11

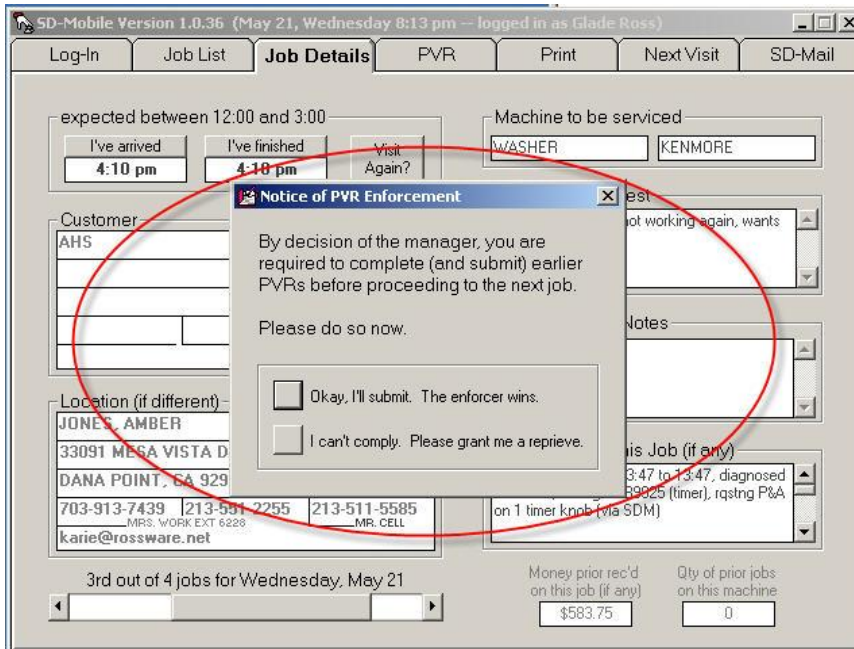
Enforcing PVRs

Sad to say, there are some techs for whom the word "scofflaw" might be deemed a compliment. These are techs that don't always do precisely what they should, including the completion of PostVisitReports in real-time, as they complete each job and before proceeding to the next. We've added a system to help discipline this variety of tech.

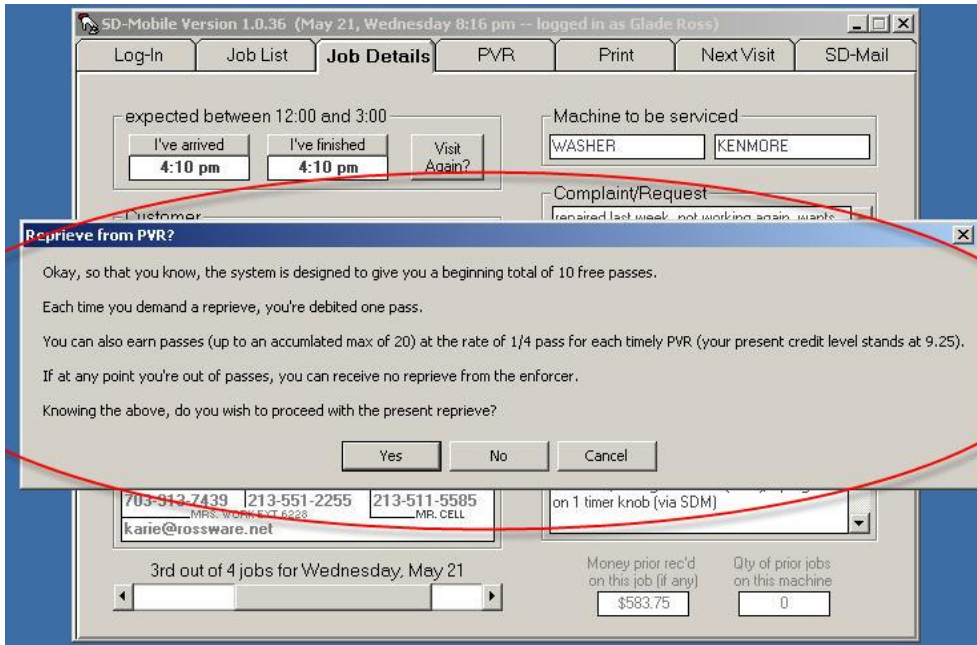
Specifically, in the SD-MobileLink program you'll see an option labeled "*Disallow next job access until prior PVR is completed.*" The general idea is, if the tech can't see info

for his next job, he's forced to do the PVR on the one he's presently finishing. Sensibly, this enforcement does not begin (for any given day) until after the tech has clocked in arrival at his first job. Thus, prior to that initial clock-in, he's permitted to review all details of all jobs.

Of course, the system is built to recognize the fact that any number of extenuating circumstances may arise, for which exceptions are required. Thus, if the tech wants to proceed to the next job and has not completed a prior PVR, he's presented with the following dialog:



If he chooses the second option, the next message is as follows

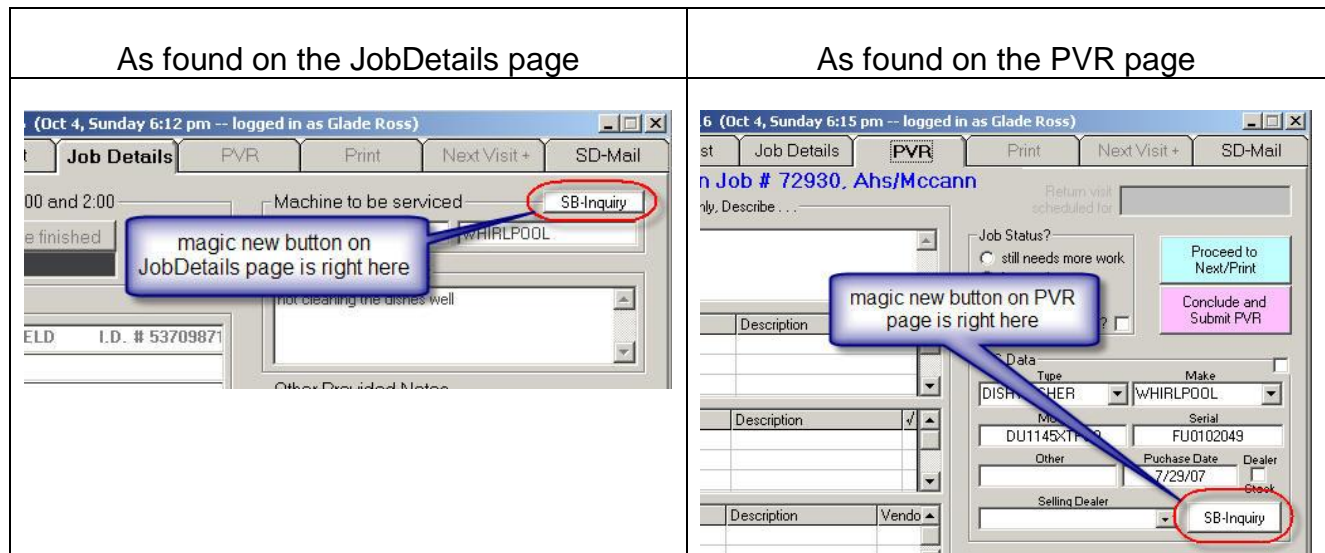


As you can see, the system provides needed flexibility, while nevertheless giving the tech heavy motivation (and, ultimately, compulsion) to do the PVRs in a timely manner.

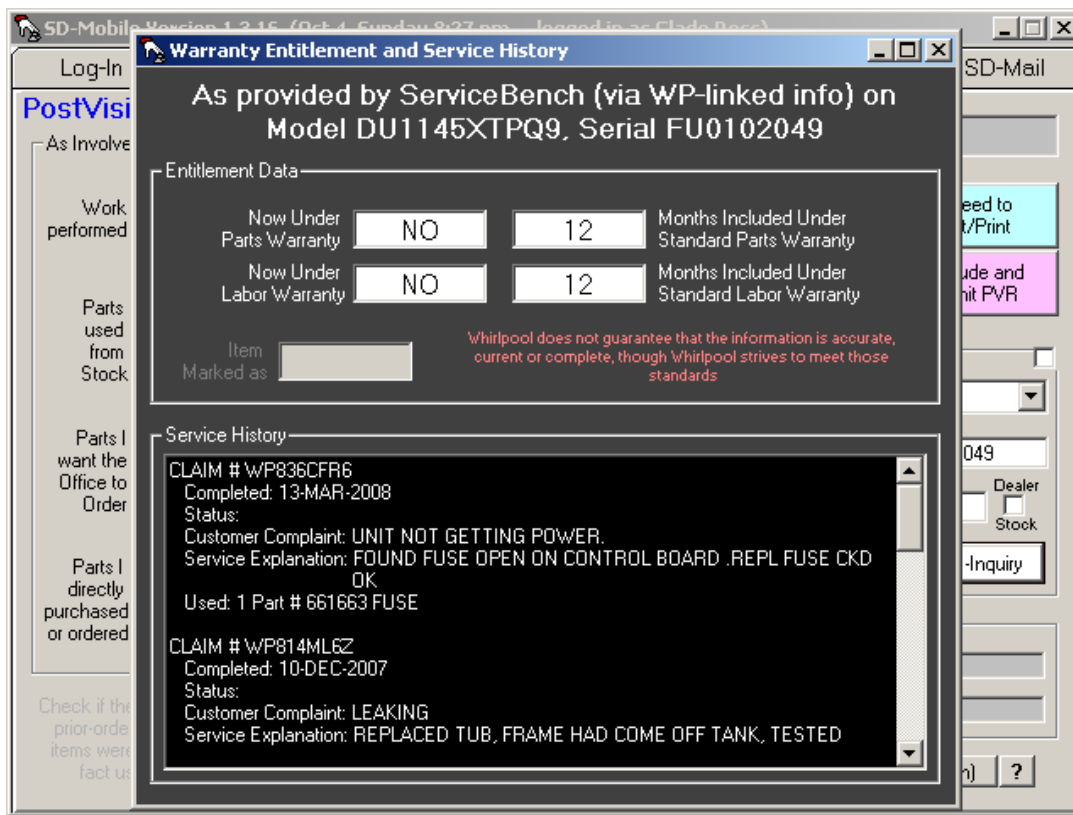
Chapter 12

Warranty-Entitlement and Product-History Inquiries

In September of '09 we worked with Whirlpool and ServiceBench to bring Whirlpool warranty servicers the ability to do a single-click inquiry to determine warranty-entitlement and product history on any particular model/serial combination. In works in ServiceDesk (from the UnitInfo form), and in Mobile via buttons provided on both the *JobDetails* and *PVR* pages:



Assuming both model and serial are present, you should find both buttons have been activated. A simple click sends a query to ServiceBench, and within less than a second you should see a result, similar to the following:



One caveat is you must be a Whirlpool warranty servicer to make this work, and each Mobile unit must be equipped with appropriate ServiceBench LogIn credentials. This *should* happen automatically, and without any tech ever seeing those credentials. At least, assuming that the office computer that's running MobilLink is, itself, equipped with the credentials.

To assure this is so, simply test the Warranty-Entitlement feature, within ServiceDesk, from that station (go to the UnitInfo form, Shift-F12, and click on the *ServiceBench Inquiry* button). If credentials are not already properly setup at that station, it will prompt you to do so.

Chapter 13

Keeping Up-To-Date

There are two aspects in staying up-to-date with these systems. One is assuring you are using the latest version of the two programs (Mobile and MobileLink). The other is assuring you (and your personnel) stay abreast of the latest features and how to benefit from them. We've done our best to help you with both tasks.

In regard to updating, both programs have a simple button you can click on to check for updates. Behind the scenes, here at Rossware, we have the ability to set an update as “Recommended” or not. Typically, if we’ve just done an update that has several major changes (which means higher risk we might have broken something), we’ll leave the recommended switch turned off—until the update has proven itself over the course of a few days (there are always intrepid users who are to try a new release regardless). After such proof, we’ll turn the switch to on, at which point the program will prompt the user to allow an update. We highly recommend, if you or your techs are prompted to allow an update, by all means DO IT!

In regard to staying abreast, this handbook will be occasionally be updated, so it’s not a bad idea to review it at least occasionally. More specifically, though, we have an on-line blog that describes new features and improvements as we ad them. It’s called the *SDM-WorkDiary* (cousin to the *SD-WorkDiary*). It’s easy to bring up whenever you wish. There’s a link to it in the SD-Mobile page on our website, and here’s a link for you presently: <http://rossware.net/SdmWorkDiary.htm>. Even better, both Mobile and MobileLink have a simple button on which you may click to quickly display this important, news-bringing document. We suggest doing a quick perusal as each update is performed—to see what improvements you should be poised to enjoy.

There is also some degree of importance, as we’ve often found clients who were failing, over a significant period, to enjoy something they really would have loved—simply because they did not know it was there.

Chapter 14

Usage Fees

The rate structure for use of SD-Mobile is very simple: \$10/mo for each tech whose credentials are present in your on-line TechRoster (or \$5 more if you’re using the Apple version).

There’s a minimum of \$20/mo, so if you have only one tech using, you’ll pay at the two-tech rate regardless.

On a monthly basis), we’ll email you a statement listing the techs we show as having been on your roster during that month, and indicating total charges (which we’ll auto-charge using the same method as used for your monthly support).¹²

Given the above, please be sure to refrain, in the ServiceDesk *Settings* form, from designating any techs as “*Using SD-Mobile*” unless they are truly doing so. Likewise, please be sure to remove any techs from such listing if they are no longer using Mobile. If they’re in your Settings form roster and you have them designated, SD-MobileLink will upload their credentials, our automated system will see them, and you’ll be charged.

¹² We do allow a little leniency on this. If, as an example, you initially setup some fake techs for testing purposes (and that’s all their uploads were created for), you can respond to the statement by informing us of same, and we’ll happily to remove any connected charges for that period. The one thing we ask is that you not do this to us repeatedly. Once or twice is fine, but since any such fixes require manual entry by us, we do not want them to be part of a continued and regular pattern.