

## Considerations for Mobilizing your Lotus Notes® Applications

**John Kingsley**  
Teamstudio Technical Director



**teamstudio®**  
Get the most from Notes®

World Headquarters  
Teamstudio, Inc.  
900 Cummings Center  
Suite 326T  
Beverly, MA 01915  
Sales: 800-632-9787  
Phone: 978-232-0145  
Fax: 978-232-0146  
contactus@teamstudio.com

EMEA  
Teamstudio Europe Ltd.  
United Kingdom  
Phone: +44 (0) 1480 424600  
Fax: +44 (0) 1480 411794  
contactus@teamstudio.com

France  
Teamstudio Europe Ltd.  
Phone: +33 (0) 140 21 42 34  
Fax: +33 (0) 140 21 42 35  
contactus@teamstudio.com

APAC  
Teamstudio Japan K.K.  
Phone: +81 (0) 3 6230 0810  
Fax: +81 (0) 3 6230 0811  
contactusjapan@teamstudio.com

### Why Mobilize?

It all started with email. Not any one email in particular, just the fact that you could get it on your smartphone.

But now that “there’s an app for that,” you probably want employees to use their mobile devices to do more than just check and send email—after all, you have to justify that \$100-a-month data plan fee somehow.

Mobile devices are becoming ubiquitous these days, and the market is continuing to grow very quickly, so it’s probably safe to assume that your company is using smartphones in some capacity or another. But are you using them to their fullest potential? If not, why not?

Think about it: You almost always have a cell phone signal, but you usually have to search to find a wi-fi hotspot. So your laptop doesn’t always do you any good, even though it’s technically a “mobile” device. In addition, for everyday work activities, I personally am more likely to bring my BlackBerry® with me than I am to lug my laptop around. I’m guessing you would prefer to do the same.

Has it ever occurred to you to pull out your laptop while waiting in the airport security line? Me neither. So, here’s a scenario for you:

One of your team members has just finished up a project at a client site and is waiting to go through security at the airport. Instead of getting into a staring contest with the four-year-old kid in line in front of him, he could actually make productive use of his time.

What if a business-critical application he uses every day was mobilized and available on his smartphone? For example, he could pull up a project tracking application and update the status of the project he just completed. Sure, he could send an email, but I’m willing to bet he’d rather just quickly access the document, make his update, and be done with it. Having remote access to mobile applications would enable him to be more responsive to his many stakeholders: his boss, his customers, the accounting department, and on and on. Pretty cool, no?

One thing to keep in mind is that mobilizing a Domino® application is more complicated than simply Web-enabling an application. When building an application that’s accessible via the Web, all you have to think about is making it compatible with the various Web browsers in use. When mobilizing your application, there are just as many Web browsers to worry about—and there are even more options, in addition to Web browsers, for making that information available on a smartphone. And never mind the miniscule screen!

So where do you start?

## What to Mobilize

First, you must understand that when users say they want to access an application from their phone, they don't mean they want access to the company's entire CRM database, tech support knowledge base, or project management system.

In most cases, it would make sense for them to perform a very small subset of functions—without having to fire up their laptop and hope for a wi-fi hotspot. So, your first challenge is to identify the functions or operations that would be good candidates for mobilization.

In the pages that follow, I'll discuss three main functions that would be great to mobilize: Simple approvals, form fill, and referencing information.

### Workflow and Simple Approvals

In many companies, reviews and approvals are embedded within the project management process. Approvals must be obtained before a project can advance to the next stage. These are excellent processes to move to a mobile device, because the request has already been completed, so the data entry requirements are fairly small.

Several people we've talked to have processes that include complicated documentation that must be reviewed as part of the approval process. In general, the more supporting documentation there is, the longer the project has been around. You are probably not often notified about a new project in an email that includes stacks of supporting documentation, along with a request for approval. So, how much information actually needs to be included in the approval request? And do you really want to view it on a 2-inch screen?

Here at Teamstudio, we have an internal process in which requests for approval are automatically rejected if the request is also the first notification of a new project. The project should already be on the table and forecasted before requests for approval are made.

### Form Fill

Time sheets, expense reports, vacation requests, and incident reports are examples of applications in which a form needs to be completed, but may not require a lot of data. Your incident report may be more complicated than mine, but you get the idea.

In these cases, simply pulling up the form on your smartphone is much more efficient than starting up the laptop, opening the application, making the changes, and saving your updates. However, keeping the forms as simple as possible is key—you want to be kind to your users and reduce the amount of typing on those tiny smartphone keypads.

### Reference Application

Ideal for traveling employees, this could be a simple list of jobs to perform that day, or a customer contact history for a regional sales rep, or maybe even a customized contact list for a particular area. Because the data is relatively static, but remote access is required, these kinds of applications are great choices for mobilizing. And, the ability to have the required information at one's fingertips is invaluable to road warriors.

## How to Mobilize

There are a few options here, but keep in mind that none of them is a magic bullet. With any of these choices, you will have to write code or change some part of your process to incorporate mobile users into your infrastructure. In addition, the option you choose will depend largely on the requirements of your application.

Here are a few questions to keep in mind as you begin to evaluate the various options for mobilizing your apps:

- Does it require real-time access to data?
- Do users need access to data, regardless of whether or not they have network connectivity?
- Does your development staff have the skills to support the (inevitable) changes to the application?
- What are the security implications?

Now that you've got all that in mind, let's take a look at your choices.

### **Web-enable an Application, Use the Browser on the Smartphone**

There are many ways to go about Web-enabling your apps, so in all likelihood your development team wouldn't need extensive training to learn new skills or coding language. However, this method also comes with significant challenges:

- Remember that tiny smartphone screen? For obvious reasons, you will need to resist the temptation to display all 6 million available colors. Simpler is faster, and faster is better.
- You will be far better off putting up a separate web page for mobile users, rather than trying to figure out which browser is being used so you can serve up different content
- Because of the limited navigation capabilities of most smartphones, buttons are hard to click on. So be kind to your mobile users, and keep buttons to a minimum.
- Your users will have real-time access to data, but the data will not be accessible if there is no network connection
- As for security, no data is stored on the device, but Web browser access to the data is not one of the most secure approaches (it's just an easily crackable user name and password away)

### **Field Mapping Application**

With a field mapping approach, where you pick and choose the fields to display on the smartphone, your team will (usually) not have to write any code, and in some cases they can even put data directly on the phone. Storing data on the phone provides a much better response time than accessing apps via the Web, because the device doesn't have to go back to the Web host for each page of data.

However, keep in mind that although this approach doesn't involve writing code, you will need to change business processes, such as workflow or approvals, because there is no business logic involved with the field mapping approach. Your developers will also have less control over formatting as the mapping application picks and chooses how to display the data on the phone.



An additional consideration for putting the data on the device is the security of the removable storage. To address this concern, there is an option to encrypt the data.

Having said that, the field mapping approach has several sub-options. In some cases, the technology is complicated enough that you must use the vendor's resources to get the mapped application on the device. Others let you use your own resources.

### **Native Application**

A native application has the benefits of both storing data directly on the mobile device, and providing your developers with complete control of formatting and automation.

Using this approach provides offline access to the data when no network is available, and it's as secure as the removable storage. Note that a BlackBerry Enterprise Server can remotely erase all files on the device, and logins to the device can provide additional security.

The fit, in terms of development skills, will depend on your resources. Your team may have to learn a new development environment, or engage in a lifelong dependency on an outside resource to support it. These are significant drawbacks, especially if multiple brands of smartphones are used in your organization.

In addition, integrating the application into the rest of your infrastructure can be quite challenging (and expensive, if outsourced). One client told me that a consultant gave him a quote of 500 hours to mobilize a single application using this approach!

### **Teamstudio Unplugged™**

Teamstudio Unplugged™ is an approach that combines the best of a native application and field mapping. It puts a Notes® thin client on the smartphone, and the database is simply replicated to the device. This means your end users get their data fast, because it's stored on the device for optimum response times.

Unplugged provides access to information as old as the last replication, but it also provides offline access when there is no network connectivity.

Because Unplugged is a replica of a Notes database, it's very easy to integrate into your infrastructure. It also integrates into the native BlackBerry menu system, so users have a familiar way of navigating to other views or adding automation to the application on the device. And, access is as secure as it is in a native application, because it uses the same established and certified AES security you already use for email transfer.

In terms of design, your development team can use the standard Domino® design elements they're already familiar with, which gives them control over formatting and automation. In addition, if they're familiar with the latest versions of Domino, they'll use their existing skills to mobilize applications quickly and efficiently—keeping the



learning curve to a minimum.

Something else to consider is that, if you choose an approach that puts your data on the smartphone, like Unplugged, you'll need to think about the size of the documents and the amount of data you want to load onto the device.

A typical Notes document needs a lot of information for the workflow component, but after considering the function you want to mobilize, the amount of data sent to the device can be greatly reduced. In one recent project, our client's original documents weighed in at 95KB. After reducing the fields to only what was needed, they were cut down to under 20KB.

### Where it's Mobilized

A final consideration for mobilizing your applications is where the automation actually takes place. Your current implementation probably does a lot of things when a user clicks a button. As such, if the data is stored on the device, your LotusScript® functions probably won't work the way they do for you now. One way to handle this is to schedule the agent to run after documents are created or modified post-replication.

With the Teamstudio Unplugged approach, documents changed on the smartphone are replicated back to the server, and the agent completes the logic formerly processed as a result of the button click. The updated document is then deleted so it can be removed from the smartphone on the next replication.

### Now it's Time to Free Your Apps!

In this paper, we've reviewed several viable options for mobilizing your Notes applications, and only you and your team can decide what will work best for your organization. One thing, however, is clear. When you give it some thought, you'll find that mobilizing carefully selected applications and functions makes good business sense.

As we've discussed, extending your Notes apps into the field through mobilization will not only help improve your organization's productivity—it'll also go a long way to justifying that pricey data plan.

To sum up, a word to the wise: Whatever method you choose, once you start to put applications on the smartphone, perfect candidates for mobilization will start popping up everywhere! Consider yourselves warned.

© Copyright 2010 Teamstudio, Inc. All rights reserved. Teamstudio and Teamstudio CIAO! are registered trademarks of Teamstudio, Inc. The IBM logo and the Business Partner emblem are trademarks of International Business Machines Corporation in the United States, other countries, or both. All other trademarks and registered trademarks are the property of their respective owners.

