

## Handling Faxes and 2D Barcodes in Coldfusion 8

Posted At : December 12, 2007 11:10 AM | Posted By : Mark Kruger

Related Categories: Coldfusion 8

I have a secret that has nothing to do with underwear. People think I'm pretty smart but the truth is that I'm surrounded every day by talented developers who know how to solve problems and find innovative solutions. Just rubbing shoulders with people like that tends to increase my level of knowledge and ability. Among those people is **Ryan Stille**, a Coldfusion and Linux Guru who has added tremendously to the wealth of knowledge here at CF Webtools. In the last few days Ryan has managed to piece together a cost effective and innovative solution for a customer who wishes to handle incoming faxes in Coldfusion. Eventually he will post a CFC on his blog - but meanwhile I wanted to explain the problem and the elegant solution involved.

### Just the Fax Ma'am

A customer of ours has subscription users who log into his system. The customer wanted his users to be able to get documents into their account using fax. Somehow the arriving faxes (in digital form) would be read by the system and placed in the users account.

We set up VOIP trunks into our asterisk server so that incoming faxes arrived as PDF files. So far so good - we had the faxes in digital format. The next hurdle was how to figure out *who the user was* from the cover sheet. There were a variety of hurdles to overcome.

- **File size and Images** - Even though we were only concerned with the first page the PDF was basically  $n$  number of pages each containing what amounted to an embedded tiff or jpg file. How do we "get at" just the first page.
- **Information Encoding** - What do we put on the cover sheet that is machine "readable" and will survive the fax process. Faxes are dirty, inexact machines. The result is often quite imperfect.
- **Cost and Time** - We found some commercial Optical Character Recognition software (OCR) on the market along with some OCR open source projects. Nothing we looked at seemed cost effective or simple enough to make for an elegant solution.
- **Weight Gain** - I have been gaining weight. Ok, this is really not a problem with our fax solution but I thought I'd throw it in here to see if you are paying attention.

Coldfusion developer **Jim Rising** suggested on the **CF-Talk** list that we look into using a 2D bar code. If you are not familiar with a 2D barcode (or a "data matrix" as it is sometimes called), it looks like this:

Jim also pointed us to a company called **Java4Less**, which had some commercial components for less than 120.00. We decided to experiment to see how easy it would be to use them in Coldfusion. Before Ryan tried his hand I came up with a few attempts using the sample Java application I found along with the docs for the object. The problem I ran into was that I could not get a "bufferedImage" object populated to

use in creating an "RImage" object (a specific custom class needed by the reader).

I won't give away Ryan's thunder but let's just say that he was able to overcome this hurdle and several others (including some very tricky issues with channels) using CF8's new "CFIMAGE" tag and the various image functions now native to Coldfusion:

- ImageGetBufferedImage( ) - not surprisingly, this function fixed my problem with getting a Buffered image object populated.
- ImageGetWidth() and ImageGetHeight()
- ImageCopy()
- ImageNew()
- ImagePaste() - I had no idea we would ever need this one.

It almost seemed like Coldfusion 8 had anticipated our need to accomplish this very task.

## Image Thumbnail

One more little tip I'll leave you with. Remember how one of our hurdles was figuring out how to get at the embedded image on the first page. The new "Cfpdf" tag came to our rescue. Since Ryan knew that the barcode was on the first page of the fax, he simply created a thumbnail of the first page using CFPDF. The code looks like this:

```
<cfpdf      action="thumbnail"
           source="/tmp/barcode.pdf"
           destination="/tmp/pdfthumbs"
           scale="100"
           resolution="high"
           pages="1"
           format="png"/>
```

This code made a PNG file 100 percent in size that could be read by our reader class.

## Summary

To recap our solution, using a combination of Coldfusion 8's image functions, Cfpdf, and CFImage combined with two commercial jar files from [Java4Less](#) we were able to write 2D bar codes encoded with user account tokens onto a printable cover sheet, and then read the 2D bar codes from incoming faxes. This allowed us to deliver faxes directly to a users account. I'd have to say that rocks.

Stay tuned. When Ryan finishes the CFC code he will post it for download. Using the CFC and a licensed copy of the datamatrix and RVision objects you will be able to handle 2D encoding and decoding from within Coldfusion.