

techno.blog("Dion")

Jun 10

smtp2web.com: Bridge SMTP to HTTP; Let App Engine accept Email

Google, Tech with tags: appengine, Python

[ShareThis](#)

Nick Johnson, of Google, has created a nice bridge service [smtp2web.com: Allow App Engine apps to receive email](#).

This is perfect timing, as after my App Engine talk in Prague, a nice gent came up to me and asked for just this. He wanted to process email in his application and didn't think he could.

One shortcoming of purely HTTP-based webapps such as App Engine is that they can't receive email. I know that some people are wanting to create App Engine apps that do this, so I put together a service to facilitate this. It runs as a standard SMTP server, and sends all the email it receives via HTTP POST to a user-designated URL. It's free to use (but [not to abuse](#))

To use the service, you setup an email account (say `dalmaer@smtp2web.com`) and a URL to point too. Of course, if you want an email on your own domain you could alias `dalmaer@yourdomain.com` to `dalmaer@smtp2web.com`).

Then, you would configure an App Engine controller to receive the email contents:

```
from google.appengine.ext import webapp
import email

class EmailHandler(webapp.RequestHandler):
    def post(self):
        sender = self.request.GET.get("from", "")
        recipient = self.request.GET.get("to", "")
        message = email.message_from_string(self.request.body)
        # Do stuff with the email message
```

The code for the service has also been [released as open source on Google Code](#) so you can check it out. You will find that it runs as a Twisted application:

```
from twisted.application import service
from twisted.application import internet
from twisted.enterprise import adbapi

import sys
import os
sys.path.append(os.path.dirname(__file__))
```

smtp2web.com: Bridge SMTP to HTTP; Let App Engine accept Email on Dion Almaer...

<http://almaer.com/blog/smtp2webcom-bridge-smtp-to-http-let-app-engine-accept-email>