



POP3による メール受信機能を 実装する

サーバーからメール取得までの流れを プログラミング

大澤 文孝 OSAWA, Fumitaka

はじめに

メールの送信ができたところで、次に、メールの受信部の作成に入ります。

メールを受信するためのクラスは、.NET Frameworkには用意されていません。そのためTCPで通信してメールを受信するクラスを自分で作らなければなりません。

メールを受信するプロトコルとしては、POP3とIMAP4がありますが、ここでは、実装が比較的簡単なPOP3を

扱います。

POP3のやりとり

POP3は、RFC1939に記載されています。

POP3クライアントを作るためには、まず、プロトコルの理解が必須です。

図1に、POP3におけるやりとりを示します。まずは、この図を見ながら、どのようなやりとりが必要なのかを説明します。

なお、前稿でも触れたように、実際のメールは、日本語の場合には、ISO-2022-JPで記述され、またさらに、Quoted-PrintableやBase64でエンコードされていることもあります。しかし本稿では、話を簡単にするため、エンコードは無視します。

エンコードについては、別稿でまとめて説明することにし、本稿では、ともかくメールを受信する機能までを実装することにします。

コマンドの送信とその応答の基礎

POP3では、コマンドは1行で完結します。用いられる改行コードは「CRLF」です。

表1に、POP3で利用できるコマンドを示します。

POP3のコマンドは、大文字と小文字の区別はありません。

応答行

サーバーの応答は、「1行の場合」と「複数行の場合」があります。

レベル >>> Level

1 2 3 4 5

サンプル >>> Sample

この記事で取り上げたソースコードおよびサンプルプログラムは、<http://www.shoeisha.com/mag/windev/>からダウンロード可能です。

言語 >>> Language

Visual C#

ツール >>> Tool

Visual C# 2005 Professional



図1：POP3におけるメール受信の流れ



いずれの場合にも、1行目は、「+OK」か「-ERR」のいずれかで始まります。

- ・ +OK
コマンドが成功したことを示します。
- ・ -ERR
エラーであったことを示します。

「+OK」や「-ERR」に続き、ひとつ空白が入って、その意味を示す文字列や結果の値が続きます。

たとえば、

PASS <パスワード>

のように、PASSコマンドを使ってパスワードを送る場合、パスワードが正しいければ、

表1：POP3のコマンド

コマンド	意味
STAT	メールの数とバイト数を得る
LIST	メールの一覧を得る
RETR	メールを受信する
DELE	メールを削除する
NOOP	何もしない(タイムアウトによる切断を避けるために定期的に送信するなどの用途に使われる)
RSET	ステータスをログイン直後に戻す (DELEで削除したメールが復帰する)
QUIT	切断する

以下はオプションなので、実装されていないPOP3サーバーも存在する

コマンド	意味
USER	POP認証のユーザー名を指定する
PASS	POP認証のパスワードを指定する
APOP	APOP認証のユーザー名とMD5文字列を指定する
TOP	RETRと同様にメールを受信するが、「ボディ部」を何行読み取るのかを指定できる。ボディ部を0行にすればヘッダ部だけを読める
UIDL	それぞれのメールに割り当てられた固有の識別子 (POP3サーバーにログインするたびに变化しないもの) を取得する