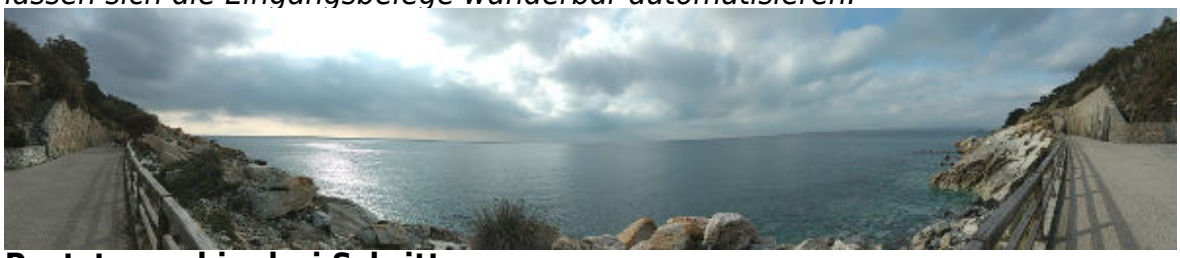
ArchivistaBox 2017/III: Posteingang vollautomatisiert

Egg, März 2017: *Mit der Version 2017/III steht das Abstempeln der eingehenden Post zur Verfügung. Wurde bisher das Datum des Posteinganges im Feld ‚Datum' erfasst, so kann neu eine automatisierte Haftnotiz erfasst werden, die auf dem Beleg automatisiert angebracht wird. Ganz egal, ob gescannte Dokumente oder digitale Dateien, mit den neuen Stempel-Funktionen lassen sich die Eingangsbelege wunderbar automatisieren.*



Poststempel in drei Schritten

Mitunter sind Kunden und Interessenten erstaunt, wie einfach und elegant neue Funktionen realisiert werden. Dies ist beim neuen Poststempel nicht anders.

1. Schritt: Skript erstellen

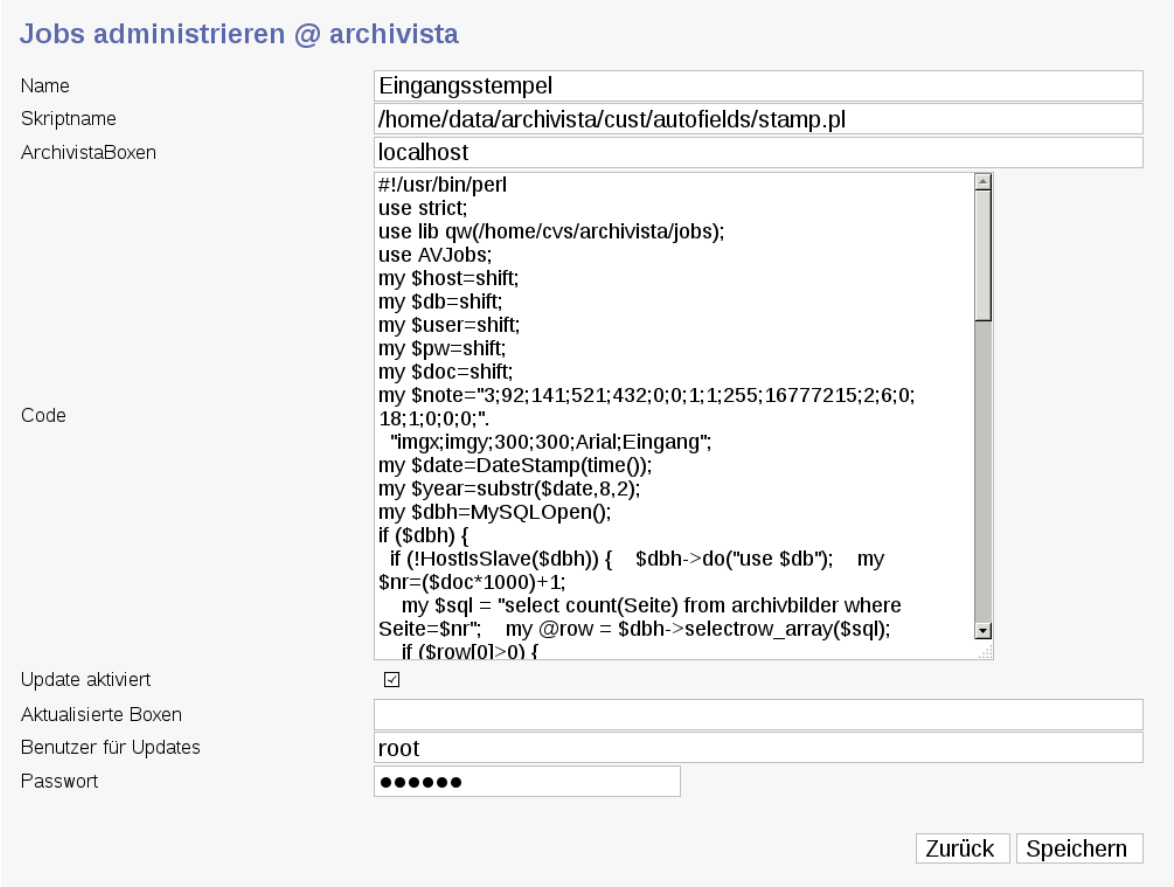
Dank der vollintegrierten Skriptfähigkeit von ArchivistaDMS wird für das Anbringen von Eingangsstempeln einzig ein kleines Skript benötigt:

```
#!/usr/bin/perl
use strict;
use lib qw(/home/cvs/archivista/jobs);
use AVJobs;
my $host=shift;
my $db=shift;
my $user=shift;
my $pw=shift;
my $doc=shift;
my $note="3;92;141;521;432;0;0;1;1;255;16777215;2;6;0;18;1;0;0;0;".
  "imgx;imgy;300;300;Arial;Eingang";
my $date=DateStamp(time());
my $year=substr($date,8,2);
my $dbh=MySQLOpen();
if ($dbh) {
  if (!HostIsSlave($dbh)) {
    $dbh->do("use $db");
    my $nr=($doc*1000)+1;
    my $sql = "select count(Seite) from archivbilder where Seite=$nr";
    my @row = $dbh->selectrow_array($sql);
    if ($row[0]>0) {
      $sql = "select Notes from archivseiten where Seite=$nr";
      my @row = $dbh->selectrow_array($sql);
      if ($row[0] eq "") {
        $sql = "select BildInput from archivbilder where Seite=$nr";
        @row = $dbh->selectrow_array($sql);
        if (length($row[0])>0) {
          my $imgo = ExactImage::newImage();
          ExactImage::decodeImage($imgo,$row[0]);
          my $width = ExactImage::imageWidth($imgo);
          my $height = ExactImage::imageHeight($imgo);
          if ($width>0 && $height>0) {
            $note =~ s/imgx/$width/g;
            $note =~ s/imgy/$height/g;
            $sql = "show tables like 'stamp$year'";
            @row = $dbh->selectrow_array($sql);
            if ($row[0] eq "") {
              $sql = "create table stamp$year ".
                "(id int(11) primary key auto_increment,Laufnummer int(11))";
              $dbh->do($sql);
            }
            $sql = "insert into stamp$year set Laufnummer=$doc";
            $dbh->do($sql);
            $sql = "select id from stamp$year where Laufnummer=$doc";
            @row = $dbh->selectrow_array($sql);
            my $count = $row[0];
            if ($count>0 && $count<=9999) {
              my $countnew = ($year*10000)+$count;
```

```
        my $note1 = $note." ".DateStamp(time())." ".$countnew;
        my $seite = ($doc*1000)+1;
        $sql = "update archivseiten set Notes='$note1' where ".
          "Seite=$seite";
        $dbh->do($sql);
        $sql = "update archiv set intRechnungNr='$countnew',".
          "DokTyp='Eingangsrechnung' where Laufnummer=$doc";
        $dbh->do($sql);
} } } } } } }
```

2. Schritt: Skript in WebAdmin erfassen

Dieses Skript fügen wir nun in WebAdmin unter ‚Jobs administrieren' hinzu, dazu der nachfolgende Bildschirm:



Dabei gibt es zu beachten, dass Skripts nur nach ‚/home/data/archivista/cust/autofields' abgelegt werden können. Zudem ist ein gültiger Benutzername und das dazugehörende Passwort festzulegen, damit das Skript in den gewünschten Ordner kopiert werden kann. Die Option ‚Update aktiviert' sowie ein gültiger Benutzername sowie das dazugehörende Passwort sind ebenfalls analog zu oben einzutragen.

Nun ist das Skript in WebAdmin bei ‚Scannen' in der gewünschten Scan-Definitionen bei ‚Vorgabfelder' einzutragen, diesmal freilich ohne den Pfad (konkret nur 'stamp.pl').

3. Schritt: Dateien verarbeiten

Sobald nun eine Datei bzw. gescannte Seiten mit der gewünschten Scan-Definition verarbeitet werden, wird das Skript jedes Mal automatisiert im Hintergrund ausgeführt.

Als Resultat wird vorliegend der gewünschte Stempel auf dem Beleg angebracht:



Damit ist der automatische Stempel beim Posteingang aktiviert. Im Unterschied zu den normalen Haftnotizen kann der Stempel zwar verschoben, nicht aber gelöscht werden.

Fazit: Automatisierung leicht gemacht

Die hier vorgestellte Stempel-Lösung steht seit einigen Monaten bei Kunden im Einsatz. Vorliegend mussten die Haftnotizen so erweitert ewrden, dass automatisiert hinzugfügt Haftnotizen nicht gelöscht werden können. Dadurch ist zwingend minimal die ArchivistaBox 2017/I erforderlich.

Die vorliegende Lösung ist überdies ein gutes Beispiel, wie einfach (und elegant) die ArchivistaBox erweitert werden kann. Dies muss nicht (wie obenstehend) mit Perl realisiert werden, denn auf der ArchivistaBox stehen neben Perl die Skriptsprachen PHP und Python ebenfalls zur Verfügung.

Es versteht sich von alleine, dass niemand selber ’skripten‘ muss (wohl aber darf), eine Kontaktaufnahme und Terminabsprache genügt, damit die Lösung für Sie realisiert werden kann.

Zum Abschluss ein Tipp: Die Stempel erscheinen auch auf den exportieren PDF-Dateien, nicht aber wenn Sie auf ‚PIC‘ oder ‚IMG‘ in der Tabellenansicht klicken, bei den letzteren beiden Optionen werden die gerasterten Originaldaten bereitgestellt, d.h. die verarbeiteten Archivdaten werden durch die Stempel nicht ‚geschädigt‘. Dagegen werden die Stempel bei den PDF-Dateien in die Bilddaten ‚gerastert‘, d.h. in den PDF-Datien können die Stempel nicht verändert (auch nicht verschoben) werden.