

Einführung in SFP
Version 1.0 draft

Steffen Köhler

1. Januar 1999

Inhaltsverzeichnis

1	Vorwort	2
2	Für den Benutzer	3
3	What are SFP-files	4
3.1	The SFP- fileformat	4
3.1.1	Blocks	4
3.1.2	Headersections	4
3.1.3	Headerlines	5
3.2	How to generate SFP-Files?	5
3.3	How to read SFP-Files?	6

Kapitel 1

Vorwort

Im folgenden Text werden die Prinzipien beschrieben, nach denen die unterschiedlichen Zeichnungen, mit denen wir tagtäglich zu tun haben, in Form einer einfachen Datenbank im Internet verfügbar und wieder abrufbar werden können.

Der Sammelbegriff, unter dem diese ganzen Punkte behandelt werden, lautet SFP. Die Abkürzung steht für und wird im Abschnitt erklärt.

Kapitel 2

Für den Benutzer

Um in der Datenbank die gewünschten Informationen später wiederfinden zu können, müssen die Dateien vor der Ablage erst mit den späteren Stichworten versehen werden, nach denen dann später gesucht werden kann.

Dazu gibt es zwei Möglichkeiten:

1. Automatische Stichwort- Erstellung: Dabei werden die Dateien bei der Erzeugung auf den Workstations gleich automatisch mit den Kenndaten des jeweiligen Zeichnungspackets versehen.
2. Manuelle Stichworteingabe: Die SFP- Dateien können zum Beispiel mit dem SimpleFP- Programm auch von Hand erstellt und mit Suchworten versehen werden.

Kapitel 3

What are SFP-files

SFP is the abbreviation for 'Simple File Packets'. SFP was designed¹ to

- put a collection of different files together in one general file
- add searchable keywords without modifying the original files
- generate these files easily with standard tools like scriptfiles

3.1 The SFP- fileformat

A SFP- File consists of one or several blocks:

```
Block 1  
....  
Block n
```

3.1.1 Blocks

Each Block contains

- a header section
- an empty line (<CR> or <LF><CR>, depending of the generating operation system)
- the original file

3.1.2 Headersections

The header sections contains several headerlines. Each header line is a simple ASCII- line, followed by a <CR> or <LF><CR> (depending of the generating operation system):

```
headerline 1  
...  
headerline n
```

¹Base design was taken from RFC822 and the 'ZConnect'-format

3.1.3 Headerlines

Each headerline consists of the header, the separator and the value:

HEADER:VALUE

- The header contains characters in capital letters [A-Z], numbers [0-9] or the underline _.
- The separator is the ":"
- the value may contain each character >0x31

Each header should occur only one time in a block. If a header occurs more as one time, only the last occurrence is valid.

Systemheaders

The SFP-Format requires some systemheaders. These are required in each block:

SFPID : This header is necessary to identify a SFP-file. It has to be the first header in each block. The *VALUE* shows the SFP-Version of the file.

FILE : The *VALUE* gives the filename & extension of the embedded file, but *without* the directory-path.

DIR (optional) : Reserved for further usage.

MIMETYPE (optional) : Reserved for further usage.

PGPKEY (optional) : Reserved for further usage.

PGPSIGN (optional) : Reserved for further usage.

LEN : The length of the embedded file as string in bytes

INFO (optional) : A description of the embedded file

Other Headers

Other headers are not defined yet.

3.2 How to generate SFP-Files?

The SFP-File can be written with a normal Unix- scriptfile:

```
file: sfp
>sfp importfile >output.sfp

echo SFPID:1.0
echo $1|awk '{gsub(".*/", "", $0)} {print "FILE:" $0}'
ls -l $1|awk '{print "LEN:" $5}'
echo
cat $1
```

3.3 How to read SFP-Files?

SFP-Files can read e.g. by a Perl-script:

```
file: sfp.pl  
>perl sfp.pl output.sfp
```

```
#!/usr/bin/perl -wT  
  
open(INFILE,$0);  
while (!eof(INFILE)) {  
    $line="a";  
    while ($line ne "\n" && !eof(INFILE)) {  
        $line=<INFILE>;  
        if ($line =~/LEN:/) {  
            ($header, $len) = split(/:/, $line, 2);  
            print $len;  
        }  
        print $line;  
    }  
    seek(INFILE,$len,1);  
}  
close(INFILE);
```