

JPEG2000 Packet Structure

Ejemplo:

| | | | |
|---------|---------|---------|-------------|
| FF 91 | 00 04 | 04 EC | C0 00 ... |
| \-----/ | \-----/ | \-----/ | \-----/ |
| 16 bits | 16 bits | 16 bits | N bits |
| SOP | Lsop | Nsop | Packet Body |

SOP: Start Of Packet marker.

Lsop: Length of marker segment in bytes (Not including the marker).

Nsop: Packet number (Possible values are between 0 and 65535).

Paquetes vacíos

Los paquetes vacíos que se han creado contienen 8 bytes y tienen el siguiente formato:

| | | | |
|---------|---------|---------|-------------|
| FF 91 | 00 04 | XX XX | 80 00 |
| \-----/ | \-----/ | \-----/ | \-----/ |
| SOP | Lsop | Nsop | Packet Body |

- **SOP:** Start of packet.
- **Lsop:** Length of marker segment in bytes (not including the marker).
- **Nsop:** Packet id [0..65535].
- **Packet Body:** Code-stream.
- $FF_{(16)} = \text{unsigned char sop_0} = 255_{(10)}$.
- $91_{(16)} = \text{unsigned char sop_1} = 145_{(10)}$.
- $00_{(16)} = \text{unsigned char lsop_0} = 0_{(10)}$.
- $04_{(16)} = \text{unsigned char lsop_1} = 4_{(10)}$.
- **XX XX** se corresponde con el id del paquete.
- $80_{(16)} = \text{unsigned char body_0} = 128_{(10)}$.
- $00_{(16)} = \text{unsigned char body_1} = 0_{(10)}$.

Nuevo tamaño de bloque para la lectura de paquetes

Se ha modificado el tamaño de la estructura que utilizábamos para leer los paquetes de los archivos `.j2c` para luego almacenarlos en nuestro formato interno `.j2c.cache`. Por ejemplo en la aplicación `woistocache` los paquetes se leían en bloques de 500 bytes utilizando la estructura `kdu_byte data[500]`. De modo que para un paquete de 2214 bytes tendríamos 5 bloques en nuestro archivo `.j2c.cache`.

```
precinct.id = 176: 7 2 0 5 6: 500 bytes
precinct.id = 176: 7 2 0 5 6: 500 bytes
precinct.id = 176: 7 2 0 5 6: 500 bytes
precinct.id = 176: 7 2 0 5 6: 500 bytes
precinct.id = 176: 7 2 0 5 6: 214 bytes
```

Hemos aumentado el tamaño de bloque y ahora utilizamos la estructura `kdu_byte data[524288]`. De este modo garantizamos que en nuestro archivo `.j2c.cache` vamos a tener el paquete en un sólo bloque. (Tengo que revisar cuál es el tamaño máximo que puede tener un paquete)

Warning!: Parsing SOPs

Hemos detectado un posible error a la hora de obtener los paquetes de un code-stream JPEG2000 si lo hacemos buscando el marcador SOP.

- **SOP marker**

Si solamente buscamos el valor `0xFF91` del marcador podemos obtener una interpretación errónea de los paquetes.

Ejemplo:

| | | | |
|---------|---------|---------|-----------|
| FF 91 | 00 04 | 00 FF | 91 40 ... |
| \-----/ | \-----/ | \-----/ | \-----/ |
| 16 bits | 16 bits | 16 bits | N bits |

| | | | |
|-----|------|------|-------------|
| SOP | Lsop | Nsop | Packet Body |
|-----|------|------|-------------|

En este caso el último byte de **Nsop** y el primer byte de **Packet body** forman el valor `0xFF91` y nos podría causar un error a la hora de obtener los paquetes de este code-stream.

En nuestros experimentos este caso era el que provocaba que nuestras utilidades “countsops” y “woistocache” obtuviesen un paquete de más.

- **EOC marker**

Si solamente buscamos el valor 0xFFD9 del marcador podemos obtener una interpretación errónea de los paquetes.

Ejemplo:

```

    FF 91      00 04      01 FF      D9 ...
\-----/  \-----/  \-----/  \-----/
   16 bits   16 bits   16 bits   N bits

SOP          Lsop      Nsop      Packet Body

```

En nuestros experimentos este caso era el que provocaba que nuestras utilidades “countsops” y “woistocache” interpretasen que se había alcanzado el final del code-stream antes de tiempo.

Cambios en la utilidad: countsops

Hemos modificado la utilidad y ahora para detectar que ha encontrado un paquete tiene que encontrar los siguientes 4 bytes seguidos.

```

    FF 91      00 04
\-----/  \-----/
   16 bits   16 bits

SOP          Lsop

```

Cambios en la librería jp2_area.cpp

Hemos modificado el método `void jp2_area::get_packet_data(FILE *f, packet_data& p)` de la librería `jp2_area.cpp` para evitar el problema de los marcadores **SOP**.

Ahora para garantizar que se ha encontrado un paquete tenemos que encontrar los siguientes 4 bytes seguidos.

```

    FF 91      00 04
\-----/  \-----/
   16 bits   16 bits

SOP          Lsop

```

También nos hemos encontrado con un caso que podría dar lugar a confusión, porque se puede hacer una mala interpretación del marcador **FF D9** (End Of Code-stream).

```

    FF 91    00 04    01 FF    D9 ... ..
\-----/  \-----/ \-----/ \-----/
 16 bits 16 bits 16 bits

```

```

SOP        Lsop    Nsop    Packet Body

```

Para solucionarlo hemos comprobado que después del marcador **FF D9** no hay más bytes.