

Presentation Graphic Stream (SUP files) BluRay Subtitle Format

📅 July 15, 2017 (<http://blog.thescorpius.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/>) 👤 TheScorpius666 (<http://blog.thescorpius.com/index.php/author/thescorpius666/>) 📁 Uncategorized (<http://blog.thescorpius.com/index.php/category/uncategorized/>)



(<https://i2.wp.com/blog.thescorpius.com/wp-content/uploads/2017/07/BluRayLogo.jpg>)

The Presentation Graphic Stream (PGS) specification is defined in the US Patent US 20090185789 A1 (<https://encrypted.google.com/patents/US20090185789?cl=da>). This graphic stream definition is used to show subtitles in BluRay movies. When a subtitle stream in PGS format is ripped from a BluRay disc is usually saved in a file with the SUP extension (Subtitle Presentation).

A Presentation Graphic Stream (PGS) is made of several functional segments one after another. These segments have the following header:

Name	Size in bytes	Description
Magic Number	2	"PG" (0x5047)
PTS	4	Presentation Timestamp
DTS	4	Decoding Timestamp

Name	Size in bytes	Description
Segment Type	1	0x14: PDS 0x15: ODS 0x16: PCS 0x17: WDS 0x80: END
Segment Size	2	Size of the segment

The DTS should indicate a time when the decoding of the sub picture starts, and the PTS indicates a time when the decoding ends and the sub picture is shown on the screen. DTS is always zero in practice (at least from what I have found so far), so you can freely ignore this value. These timestamps have an accuracy of 90 kHz. This means that for example, if you have a PTS value of 0x0004C11C and you want to know the milliseconds from the start of the movie when the sub picture is shown, you have to divide the decimal value (311,580) by 90, and the result is the value you are looking for: 3,462 milliseconds (or 3.462 seconds).

As you can see, there are five types of segments used in PGS:

- Presentation Composition Segment (PCS)
- Window Definition Segment (WDS)
- Palette Definition Segment (PDS)
- Object Definition Segment (ODS)
- End of Display Set Segment (END)

The Presentation Composition Segment (PCS) is also called the *Control Segment* because it indicates a new Display Set (DS) definition, composed by definition segments (WDS, PDS, ODS) until an END segment is found.

A Display Set (DS) is a sub picture definition that might look like this:

PCS	WDS	PDS	ODS	END
-----	-----	-----	-----	-----

In a DS there can be several windows, palette and object definitions, and the composition objects define what is going to be shown on the screen.

Presentation Composition Segment

The Presentation Composition Segment is used for composing a sub picture. It is made of the following fields:

Name	Size in bytes	Description
Width	2	Video width in pixels (ex. 0x780 = 1920)
Height	2	Video height in pixels (ex. 0x438 = 1080)
Frame Rate	1	Always 0x10. Can be ignored.
Composition Number	2	Number of this specific composition. It is incremented by one every time a graphics update occurs.

Name	Size in bytes	Description
Composition State	1	Type of this composition. Allowed values are: 0x00: Normal 0x40: Acquisition Point 0x80: Epoch Start
Palette Update Flag	1	Indicates if this PCS describes a Palette only Display Update. Allowed values are: 0x00: False 0x80: True
Palette ID	1	ID of the palette to be used in the Palette only Display Update
Number of Composition Objects	1	Number of composition objects defined in this segment

The composition state can be one of three values:

- **Epoch Start:** This defines a *new display*. The Epoch Start contains all functional segments needed to display a new composition on the screen.
- **Acquisition Point:** This defines a *display refresh*. This is used to compose in the middle of the Epoch. It includes functional segments with new objects to be used in a new composition, replacing old objects with the same Object ID.
- **Normal:** This defines a *display update*, and contains only functional segments with elements that are different from the preceding composition. It's mostly used to stop displaying objects on the screen by defining a composition with no composition objects (a value of zero in the Number of Composition Objects flag) but also used to define a new composition with new objects and objects defined since the Epoch Start.

The composition objects, also known as window information objects, define the position on the screen of every image that will be shown. They have the following structure:

Name	Size in bytes	Description
Object ID	2	ID of the ODS segment that defines the image to be shown
Window ID	1	Id of the WDS segment to which the image is allocated in the PCS. Up to two images may be assigned to one window
Object Cropped Flag	1	0x40: Force display of the cropped image object 0x00: Off
Object Horizontal Position	2	X offset from the top left pixel of the image on the screen
Object Vertical Position	2	Y offset from the top left pixel of the image on the screen

Name	Size in bytes	Description
Object Cropping Horizontal Position	2	X offset from the top left pixel of the cropped object in the screen. Only used when the Object Cropped Flag is set to 0x40.
Object Cropping Vertical Position	2	Y offset from the top left pixel of the cropped object in the screen. Only used when the Object Cropped Flag is set to 0x40.
Object Cropping Width	2	Width of the cropped object in the screen. Only used when the Object Cropped Flag is set to 0x40.
Object Cropping Height Position	2	Heightl of the cropped object in the screen. Only used when the Object Cropped Flag is set to 0x40.

When the Object Cropped Flag is set to true (or actually 0x40), then the sub picture is cropped to show only a portion of it. This is used for example when you don't want to show the whole subtitle at first, but just a few words first, and then the rest.

Window Definition Segment

This segment is used to define the rectangular area on the screen where the sub picture will be shown. This rectangular area is called a *Window*. This segment can define several windows, and all the fields from Window ID up to Window Height will repeat each other in the segment defining each window. You can see it more clearly in the example at the end of this page. Its structure is as follows:

Name	Size in bytes	Description
Number of Windows	1	Number of windows defined in this segment
Window ID	1	ID of this window
Window Horizontal Position	2	X offset from the top left pixel of the window in the screen.
Window Vertical Position	2	Y offset from the top left pixel of the window in the screen.
Window Width	2	Width of the window
Window Height	2	Height of the window

Palette Definition Segment

This segment is used to define a palette for color conversion. It's composed of the following fields:

Name	Size in bytes	Definition
Palette ID	1	ID of the palette

Name	Size in bytes	Definition
Palette Version Number	1	Version of this palette within the Epoch
Palette Entry ID	1	Entry number of the palette
Luminance (Y)	1	Luminance (Y value)
Color Difference Red (Cr)	1	Color Difference Red (Cr value)
Color Difference Blue (Cb)	1	Color Difference Blue (Cb value)
Transparency (Alpha)	1	Transparency (Alpha value)

There can be several palette entries, with different palette IDs so the last five fields can repeat.

Object Definition Segment

This segment defines the graphics object. These are images with rendered text on a transparent background. Its structure is explained in the following table:

Name	Size in bytes	Description
Object ID	2	ID of this object
Object Version Number	1	Version of this object
Last in Sequence Flag	1	If the image is split into a series of consecutive fragments, the last fragment has this flag set. Possible values: 0x40: Last in sequence 0x80: First in sequence 0xC0: First and last in sequence (0x40 0x80)
Object Data Length	3	The length of the Run-length Encoding (RLE) data buffer with the compressed image data.
Width	2	Width of the image
Height	2	Height of the image
Object Data	variable	This is the image data compressed using Run-length Encoding (RLE). The size of the data is defined in the Object Data Length field.

The Run-length encoding method is defined in the US 7912305 B1 patent (<https://www.google.com/patents/US7912305>). Here's a quick and dirty definition to this method:

Code	Meaning
CCCCCCCC	One pixel in color C
00000000 00LLLLLL	L pixels in color 0 (L between 1 and 63)

Code	Meaning
00000000 01LLLLLL LLLLLLLL	L pixels in color 0 (L between 64 and 16383)
00000000 10LLLLLL CCCCCCCC	L pixels in color C (L between 3 and 63)
00000000 11LLLLLL LLLLLLLL CCCCCCCC	L pixels in color C (L between 64 and 16383)
00000000 00000000	End of line

End Segment

The end segment always has a segment size of zero and indicates the end of a Display Set (DS) definition. It appears immediately after the last ODS in one DS.

Example

Let's see a real world example. Take a look at this section of a SUP file:

00348a10	50 47 05 88 fd ec 00 00	00 00 16 00 13 07 80 04	PG.....
00348a20	38 10 01 ae 80 00 00 01	00 00 00 00 03 05 00 6c	8.....l
00348a30	50 47 05 88 fd ec 00 00	00 00 17 00 13 02 00 03	PG.....
00348a40	05 00 6c 01 79 00 2b 01	02 e3 03 a0 01 d8 00 2b	..l.y.+.....+
00348a50	50 47 05 88 fd ec 00 00	00 00 14 00 9d 00 00 00	PG.....
00348a60	10 80 80 00 01 10 80 80	ff 02 1f 80 80 ff 03 2d-
00348a70	80 80 ff 04 3c 80 80 ff	05 4a 80 80 ff 06 59 80<....J....Y.
00348a80	80 ff 07 68 80 80 ff 08	76 80 80 ff 09 85 80 80	...h....v.....
00348a90	ff 0a 93 80 80 ff 0b a2	80 80 ff 0c b1 80 80 ff
00348aa0	0d bf 80 80 ff 0e ce 80	80 ff 0f dc 80 80 ff 10
00348ab0	eb 80 80 ff 11 10 80 80	11 12 10 80 80 22 13 10"....
00348ac0	80 80 33 14 10 80 80 66	15 10 80 80 99 16 10 80	..3....f.....
00348ad0	80 aa 17 10 80 80 bb 18	10 80 80 55 19 10 80 80U....
00348ae0	77 1a 10 80 80 cc 1b 10	80 80 88 1c 10 80 80 44	w.....D
00348af0	1d 10 80 80 dd 1e 10 80	80 ee 50 47 05 88 fd ecPG....
00348b00	00 00 00 00 15 21 c2 00	00 00 c0 00 21 bb 01 79!.....!...y
00348b10	00 2b 00 01 11 00 86 12	00 06 00 87 12 00 12 11	..+.....
(...)			
0034acc0	0f 11 00 86 12 00 0e 00	00 50 47 05 88 fd ec 00PG.....
0034acd0	00 00 00 80 00 00	

This is a complete Display Set. These are the segments:


- **First Segment: PCS, offset 0x00348a10**
 - Magic Number: "PG" (0x5047)
 - Presentation Time: 17:11.822 (92,863,980 / 90)
 - Decoding Time: 0
 - Segment Type: **PCS** (0x16)
 - Segment Size: 0x13 bytes
 - Width: 1920 (0x780)
 - Height: 1080 (0x438)
 - Frame rate: 0x10
 - Composition Number: 430 (0x1ae)
 - Composition State: Epoch Start (0x80)
 - Palette Update Flag: false
 - Palette ID: 0


- Number of Composition Objects: 1
- Object ID: 0
- Window ID: 0
- Object Cropped Flag: false
- Object Horizontal Position: 773 (0x305)
- Object Vertical Position: 108 (0x06c)
- **Second Segment: WDS (0x17), offset 0x00348a30**
 - Magic Number: "PG" (0x5047)
 - Presentation Time: 17:11.822 (92,863,980 / 90)
 - Decoding Time: 0
 - Segment Type: **WDS** (0x17)
 - Segment Size: 0x13 bytes
 - Number of Windows: 2
 - Window ID: 0
 - Window Horizontal Position: 773 (0x305)
 - Window Vertical Position: 108 (0x06c)
 - Window Width: 377 (0x179)
 - Window Height 43 (0x02b)
 - Window ID: 1
 - Window Horizontal Position: 739 (0x2e3)
 - Window Vertical Position: 928 (0x3a0)
 - Window Width: 472 (0x1d8)
 - Window Height 43 (0x02b)
- **Third Segment: PDS (0x14), offset 0x00348a50**
 - Magic Number: "PG" (0x5047)
 - Presentation Time: 17:11.822 (92,863,980 / 90)
 - Decoding Time: 0
 - Segment Type: **PDS** (0x14)
 - Segment Size: 0x9d bytes
 - Palette ID: 0
 - Palette Version: 0
 - 31 palette entries
- **Fourth Segment: ODS (0x15), offset 0x00348afa**
 - Magic Number: "PG" (0x5047)
 - Presentation Time: 17:11.822 (92,863,980 / 90)
 - Decoding Time: 0
 - Segment Type: **PDS** (0x15)
 - Segment Size: 0x21c2 bytes
 - Object ID: 0
 - Object Version Number: 0
 - Last in sequence flag: First and last sequence (0xC0)
 - Object Data Length: 0x0021bb bytes
 - Width: 377 (0x179)
 - Height: 43 (0x02b)
- **Fifth Segment: END (0x80), offset 0x0034acc9**
 - Magic Number: "PG" (0x5047)
 - Presentation Time: 17:11.822 (92,863,980 / 90)
 - Decoding Time: 0


- Segment Type: **END** (0x80)
- Segment Size: 0 bytes


This Display Set will show an image of 377×43 in size, starting at 17 minutes, 11.822 seconds on the screen at offset 773×108.


Share this:

 (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/?share=twitter&nb=1>)

 (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/?share=facebook&nb=1>)

 (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/?share=linkedin&nb=1>)

 (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/?share=pinterest&nb=1>)

 (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/?share=reddit&nb=1>)

Related

Pebble App Modding to Include Notification Filters
(<http://blog.thescorpious.com/index.php/2016/09/01/pebble-app-notification-filters/>)
September 1, 2016
In "Pebble"

Using Self-Signed Certificates for Amazon Alexa Skills
(<http://blog.thescorpious.com/index.php/2015/11/15/using-self-signed-certificates-for-amazon-alexa-skills/>)
November 15, 2015
In "Amazon Alexa"

A Useful Git Tutorial - Part II
(<http://blog.thescorpious.com/index.php/2015/10/25/a-useful-git-tutorial-part-ii/>)
October 25, 2015
In "git"

3 thoughts to “Presentation Graphic Stream (SUP files) BluRay Subtitle Format”



FREDRIK RENTSCH ([HTTPS://PLUS.GOOGLE.COM/116046650226532775732](https://plus.google.com/116046650226532775732))

July 13, 2018 at 4:25 pm (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/#comment-158>)

I am currently working on a script for changing the size of subtitles after cropping a video in Handbrake. This article has been very usefull, however I think you forgot to include the “number of windows” byte of the Window Definition Segment and that the windows repeat over that number. I had to look through the source code of BDSub2sub before I realized this, and after realizing this I finally understood how your WDS could have 0x13 in size which didn’t add up with the 9 bytes specified in the WDS segment.

REPLY



THESCORPIUS666

July 13, 2018 at 4:45 pm (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/#comment-159>)

You're absolutely right! Even in the example you can see that there are two windows. I'll update the post with that information. The problem is that the Patent Document doesn't specify it, because it isn't clear there.

REPLY



ZERGFTW

January 31, 2020 at 12:05 pm (<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/#comment-425>)

Useful note: PGS supports up to 64 presentation objects in one epoch. But only 2 objects can be shown simultaneously. I.e. only 2 objects can be in one Presentation Composition Segment.

REPLY

LEAVE A REPLY

Enter your comment here...

◀ Control your X10 Devices with the Amazon Echo using the Smart Home Skill API
(<http://blog.thescorpious.com/index.php/2016/11/19/control-x10-amazon-echo-smart-home-skill/>)

DONATE

Do you think what you read was useful and it helped you? Please consider a donation.



([https://www.paypal.com/cgi-bin/webscr?](https://www.paypal.com/cgi-bin/webscr?cmd=_donations&business=NVDVRMP4WTFJU&lc=US¤cy_code=USD&bn=PP%2dDonationsBF%3abt_n_donate_LG%2egif%3aNonHosted)

[cmd=_donations&business=NVDVRMP4WTFJU&lc=US¤cy_code=USD&bn=PP%2dDonationsBF%3abt_n_donate_LG%2egif%3aNonHosted\)](https://www.paypal.com/cgi-bin/webscr?cmd=_donations&business=NVDVRMP4WTFJU&lc=US¤cy_code=USD&bn=PP%2dDonationsBF%3abt_n_donate_LG%2egif%3aNonHosted)

Search

RECENT POSTS

Presentation Graphic Stream (SUP files) BluRay Subtitle Format
(<http://blog.thescorpious.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/>)

format/)

Control your X10 Devices with the Amazon Echo using the Smart Home Skill API
(<http://blog.thescorpius.com/index.php/2016/11/19/control-x10-amazon-echo-smart-home-skill/>)

Tired of Windows cmd.exe? ConEmu is exactly what you need!
(<http://blog.thescorpius.com/index.php/2016/10/21/conemu-console-emulator/>)

Pebble App Modding to Include Notification Filters
(<http://blog.thescorpius.com/index.php/2016/09/01/pebble-app-notification-filters/>)

Control your X10 Devices with the Amazon Echo using IFTTT
(<http://blog.thescorpius.com/index.php/2016/03/25/x10-amazon-echo-ifttt/>)

RECENT COMMENTS

💬 ZergFTW on Presentation Graphic Stream (SUP files) BluRay Subtitle Format
(<http://blog.thescorpius.com/index.php/2017/07/15/presentation-graphic-stream-sup-files-bluray-subtitle-format/#comment-425>)

💬 Charlie Stricklin on Control your X10 Devices with the Amazon Echo using IFTTT
(<http://blog.thescorpius.com/index.php/2016/03/25/x10-amazon-echo-ifttt/#comment-375>)

💬 Charlie Stricklin on Control your X10 Devices with the Amazon Echo using IFTTT
(<http://blog.thescorpius.com/index.php/2016/03/25/x10-amazon-echo-ifttt/#comment-374>)

💬 Jason on DD-WRT Network Stats Graphics using Grafana and InfluxDB
(<http://blog.thescorpius.com/index.php/2016/03/14/dd-wrt-network-statistics-graphics-using-grafana-and-influxdb/#comment-348>)

💬 Jason on DD-WRT Network Stats Graphics using Grafana and InfluxDB
(<http://blog.thescorpius.com/index.php/2016/03/14/dd-wrt-network-statistics-graphics-using-grafana-and-influxdb/#comment-345>)

ARCHIVES

📅 July 2017 (<http://blog.thescorpius.com/index.php/2017/07/>)

📅 November 2016 (<http://blog.thescorpius.com/index.php/2016/11/>)

📅 October 2016 (<http://blog.thescorpius.com/index.php/2016/10/>)

📅 September 2016 (<http://blog.thescorpius.com/index.php/2016/09/>)

📅 March 2016 (<http://blog.thescorpius.com/index.php/2016/03/>)

📅 December 2015 (<http://blog.thescorpius.com/index.php/2015/12/>)

📅 November 2015 (<http://blog.thescorpius.com/index.php/2015/11/>)

📅 October 2015 (<http://blog.thescorpius.com/index.php/2015/10/>)

CATEGORIES

 [Amazon Alexa \(http://blog.thescorpius.com/index.php/category/amazon-alexa/\)](http://blog.thescorpius.com/index.php/category/amazon-alexa/)

 [cross-compile \(http://blog.thescorpius.com/index.php/category/cross-compile/\)](http://blog.thescorpius.com/index.php/category/cross-compile/)

 [dd-wrt \(http://blog.thescorpius.com/index.php/category/dd-wrt/\)](http://blog.thescorpius.com/index.php/category/dd-wrt/)

 [git \(http://blog.thescorpius.com/index.php/category/git/\)](http://blog.thescorpius.com/index.php/category/git/)

 [linux \(http://blog.thescorpius.com/index.php/category/linux/\)](http://blog.thescorpius.com/index.php/category/linux/)

 [Pebble \(http://blog.thescorpius.com/index.php/category/pebble/\)](http://blog.thescorpius.com/index.php/category/pebble/)

 [qBittorrent \(http://blog.thescorpius.com/index.php/category/qbittorrent/\)](http://blog.thescorpius.com/index.php/category/qbittorrent/)

 [tutorials \(http://blog.thescorpius.com/index.php/category/tutorials/\)](http://blog.thescorpius.com/index.php/category/tutorials/)

 [Uncategorized \(http://blog.thescorpius.com/index.php/category/uncategorized/\)](http://blog.thescorpius.com/index.php/category/uncategorized/)

 [Windows \(http://blog.thescorpius.com/index.php/category/windows/\)](http://blog.thescorpius.com/index.php/category/windows/)

META

[Log in \(http://blog.thescorpius.com/wp-login.php\)](http://blog.thescorpius.com/wp-login.php)

[Entries RSS \(Really Simple Syndication\) \(http://blog.thescorpius.com/index.php/feed/\)](http://blog.thescorpius.com/index.php/feed/)

[Comments RSS \(Really Simple Syndication\) \(http://blog.thescorpius.com/index.php/comments/feed/\)](http://blog.thescorpius.com/index.php/comments/feed/)

[WordPress.org \(https://wordpress.org/\)](https://wordpress.org/)