FAST SATELLITE DATA DISSEMINATION OF LARGE JPEG2000 COMPRESSED IMAGES WITH OGC STANDARDS

J. Maso [a, *], A. Zabala [b], X. Pons [b,a], N. Julià [a]

[a] Center for Ecological Research and Forestry Applications, UAB, 08193 Bellaterra, Barcelona, Spain- (joan.maso@uab.cat, n.julia@creaf.uab.cat)
[b] Geography Department of the Autonomous University of Barcelona 08193 Bellaterra, Barcelona, Spain- (alaitz.zabala@uab.cat, xavier.pons@ uab.cat)

KEY WORDS: Wavelet, compression, web services, standards.

ABSTRACT:

This communication exposes the way that huge dataset imagery can be compressed in JPEG2000 and served to the reaction teams using OGC standards. Several web servers will be analyzed and tested (Web Map Server, Web Feature Server, Web Coverage server, etc) but special attention will be given to the new Web Map Tile Service (WMTS) standard candidate that is expected to be approved this year. WMTS exposes a structure of resolution levels that are cut in small and uniform tiles in a very similar way that JPEG2000 does. We will discuss how JPEG2000 and WMTS can be combined together and advantages and disadvantages versus a more common approach based on prerendered classical JPEG tiles. Among the different architectures that we will discuss, it is worth mentioning, on one hand, a WMTS server that uses one or several JPEG2000 images internally, but serves to a thin client (a web browser) more classical formats like JPEG, PNG etc; on the other hand, a server that extents the current WMTS standard candidate to allow JPIP requests from a thick client in a way that the server will know the list of codeblocks that a particular client was previously received and still saves in the cache. Complexity, performance and bandwidth will be taken into account on this evaluation.

1. MANUSCRIPT

1.1 Instruction

Spatial Data Infrastructures (SDI) are excellent instruments to risk assessment studies and post crisis reaction, among many other applications (Rajabifard et al., 2004). The main sources of primary information for SDI's are satellite sensors and, eventually, airborne digital cameras that can produce huge amounts of daily raster images. The International Charter Space and Major Disasters makes available these images for free in an event of a crisis. These huge images can be discovered through catalogues and have to be downloaded to the devices of the post crisis reaction teams on the ground. Bandwidth is currently a problem in many parts of the Earth and especially in a crisis situation where some data links could be down and the available communication channels can be overloaded. Image compression techniques are crucial for efficient and fast dissemination of this critical data to the reaction teams that normally use laptop computers and mobile devices. In terms of quality and compression ratio, wavelet compression algorithms, and particularly the JPEG2000 format, have a better performance than other compression techniques (Taubman, 2002). Another critical factor in a crisis situation is the use of standards that make possible to easily do tasks with any software available, especially in an environment where the operator of the system could be under stress. JPEG2000 has been standardized in the ISO 15444, so interoperability between different software implementations can be guarantied (ISO/IEC 15444-1, 2000).

JPEG2000 offers superior benefits than JPEG for a wide variety of applications. It supports lossy and lossless compression. It offers improved quality at the same compression ratio (due to it do not have 8x8 block JPEG artefacts). It has been designed to facilitate onscreen display of big imagery and has significantly improved bit-stream scalability, which is defined at the image level. However, all these benefits come at the expense of the algorithm complexity. It means that the JPEG2000 compression and decompression times are larger than those for JPEG.

The compression process is based on the discrete wavelet transform (DWT) that generate sub-bands of coefficients that are quantized, divided in precincts and collected in rectangular arrays called codeblocks that are entropy coded. A JPEG2000 file consists in a header and a codeblock array. The header contains the necessary information to describe the image and to locate the codeblocks (Christopoulos, 2000) (see figure 1).

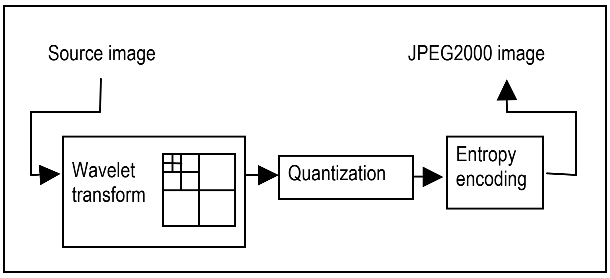

Figure 1. JPEG2000 encoding process

The standards organization Open Geospatial Consortium (OGC) has been successfully proposing several geospatial web services to interoperate with maps, features, coverages and sensors, and they are deeply accepted in the SDI community. However, some of these standards were written before JPEG2000 format started to grow and many still ignore it. Nevertheless, JPEG2000 could be used in several OGC web services to improve interoperability in the geospatial web.
Since 2000, OGC has been releasing several standards that are relevant for JPEG2000 data distribution. Particularly WMS

* Corresponding author.

specifies how to distribute fragments of maps (de la Beaujardiere 2004), Web Coverage Service (WCS) specifies how to distribute coverage images (mainly raster files) (Whiteside 2008), Sensor Observation Service (SOS) specifies how to extract information from a sensor (that could serve images) (Na 2007) and GML in JPEG 2000 for Geographic Imagery (GMLJP2) specifies how to encapsulate metadata and annotations in a JPEG2000 file (Kyle 2006). Also, OGC is in the process to approve a new standard called WMTS. This standard targets map services that are expected to have many concurrent requests. It defines some scale levels and, for each scale, it divides the database space in regular and rectangular non-overlapping blocks, called tiles, describing a pyramidal structure. These tiles could be prerendered representations of a layer (in a common image format like JPEG) or generated on-the-fly from the original dataset (that can be stored in the server as a JPEG2000 file) (Masó *et al.* 2009).

On the other hand, Part 9 of the 15444 ISO standard (JPIP) specifies an interactive protocol that allows transmitting only the needed parts (called codeblocks) of an image that a client is interactively showing on the screen. The server keeps track of all previously transmitted parts and do not send them again, because the client keeps the previous codeblocks in cache. Only never transmitted parts are sent, so that this strategy saves time and bandwidth (ISO/IEC 15444-9, 2003). This standard has the same purpose than the WMS and WMTS have, but using a totally different strategy that requires a thick client to work over and that is limited to the JPEG2000 format.

This paper exposes the way that huge dataset imagery can be compressed in JPEG2000 and served to the reaction teams in a crisis situation using OGC standards.

2. INTEGRATION IN OGC STANDARDS

2.1 Using JPEG2000 in SOS.

SOS allows requesting data from a specified measure of a specific sensor on a certain time. Particularly, the sensor could be a camera that takes pictures from time to time. The sensor can return observations in JPEG2000 format and other information using the GetObservation request. In this case, the whole JPEG2000 image will be retrieved, since GetObservation has no way to indicate an image resampling or clipping. Nevertheless, there are still advantages when using JPEG2000 formats, specially if the camera is a multispectral sensor, because JPEG2000 can compress between bands taking advantage of their redundancy and generating a smaller file.
Another approach could consist in returning the image using a JPIP service. This is possible because GetObservation request has a parameter called ResponseMode that can be set to out-of-band value to get instructions on how to get the actual observations by another protocol (Di 2007).

2.2 Using JPEG2000 and JPIP in WCS

WCS is mainly intended to serve fragments of raster files at different resolutions. JPEG2000 is an excellent format for saving the image on the server side, since is it possible to generate a fragment of the image without decompressing the whole data, with some restrictions. It is also an excellent format for transmitting the image because it achieves a high compression ratio. The transcoder determines which codeblocks are needed to extract any particular area, reads the codeblocks without uncompress and recombines them in a resulting file. Assuming that the user has also requested a JPEG2000 image, these codeblocks can be directly rearranged (transcoded) in a responded image. The problem is that this operation will only generate a resolution equal or submultiple to the original resolution of the image (in a range of resolution levels). In general the client will receive a JPEG2000 image that normally exceeds the requested area and resolution, so in order to show the image on the screen, it has to decompress each codeblock and probably clip the image and resample it to the desired resolution. This will give a client an extra work that could be avoided if the requested resolution coincides with one of the resolution levels of the original image. This will hardly be the case if the client does not receive any information about the source resolutions. Some OGC candidate profiles have explored the possibility of communicating to the client the available resolutions. Figure 2 shows a possible combination of a WCS server with JPEG transcoder and a JPEG2000 image. In this combination, the requested image is received by a WCS server that uses a transcoder to recover part of the JPEG2000 image. Then, the WCS server can choose to serve the image as is (and possibly serve it in a resolution and bounding box close to the requested ones but not equal to them) or to decompress the image, clip it and to interpolate it to the requested resolution and recompress it in a new JPEG2000 file before sending it to the client. The client can choose which process the WCS server will do using a vendor parameter.

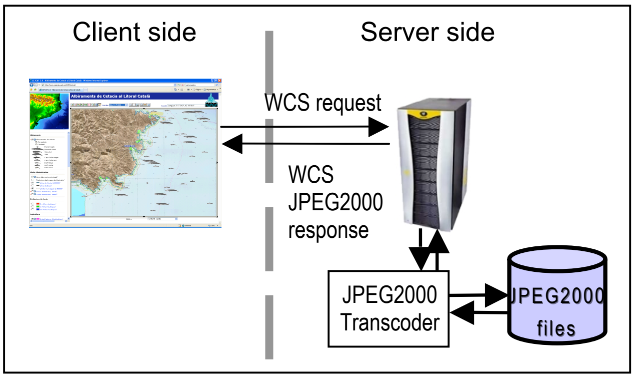

Figure 2. JPEG2000 transcoder as a base for WCS services

Figure 3 shows another possibility where a WCS client is combined with a JPIP client that communicates with a JPIP server. The advantages of this strategy are that we could use any compatible WCS client to dialog to a JPIP on the client side that maintains the codeblock cache and saves bandwidth. An extra piece of software is required to translate the WCS requests (bounding boxes and resolutions) to the JPIP client parameters (columns and rows indices and offsets).

These combinations of servers do not require any restriction on the WCS protocol because a WCS response payload is reprocessed before reaching the WCS client. To eliminate the need of reprocessing and take full advantages of the JPEG2000 format, Giacovelli suggests some small modifications on the WCS (Giacovelli 2007). He discusses the possibility of asking directly for some resolution levels instead of using WCS resx/resy, or GridOffset parameters. This will avoid the need to interpolate to a requested output resolution in the server side and will provide a faster response. This is also less flexible and makes the client more complicated.

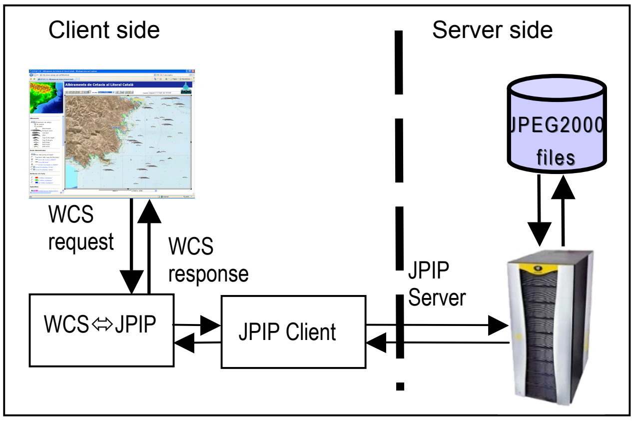

Figure 3. WCS to JPIP transformation on the client side

In the OWS-4 OGC interoperability experiment a WCS and a JPIP server were successfully combined (Gerlek 2006). The combination (see figure 4) requires 4 components: a JPIP client and JPIP server, and a WCS client and WCS server. WCS is used to discover the service and to translate the WCS request to a JPIP request. First, a WCS client requests a WCS layer specifying a bounding box and a resolution. Then, the server responds to the client with a JPIP request that the client directly submits to the JPIP server, which responds to the JPIP client that shows the image.

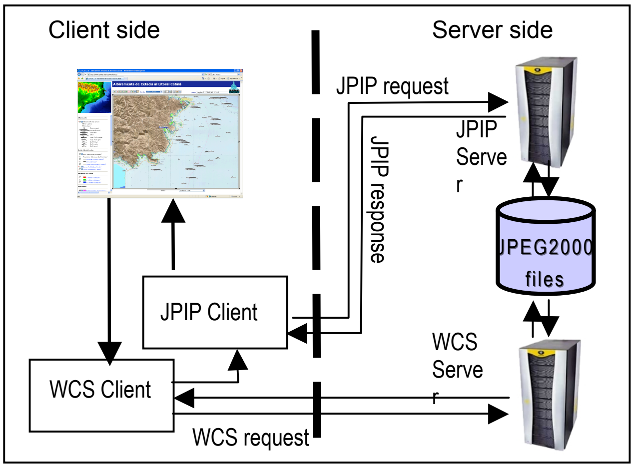

Figure 4. WCS to JPIP transformation on the client side

The OWS-5 OGC interoperability experiment defined a workflow where a sensor produces imagery that can be accessed through a JPIP server (Keens 2008). This JPIP server communicates a WCS-T that a new layer exists and transmits it to the WCS server that registers this new layer in the service metadata document. Then the new layer can be immediately retrieved from the WCS service.

2.3 Using JPEG2000 in WMS and WMTS services.

WMS is the most popular OGC standard. It defines how to communicate screen views of a map from the server to the client. A WMS GetMap request specifies the bounding box (in reference units) and the number of columns and rows of the map image. The returned map can be mostly encoded in a common format like PNG or JPEG but any other image format can be returned, like JPEG2000. GetMap operation allows requesting any image resolution so the server has to retrieve the relevant data from the original data and to render it to this particular resolution. If the server has to deal with images, it has to interpolate the image from the source resolution to the requested resolution. In the previous WCS, the response still can have the geo-reference information and metadata so the client has the need of extra data to react if the server does not respond exactly what the client has requested, but in a WMS, the server has to respond exactly what the client requires. In general a WMS server, that has data in a JPEG2000 format, has to decompress the image to the closest resolution and then clip and interpolate it to generate the exact resolution requested. Then, the server has to generate the requested image format. To generate a JPEG2000 image requires much more time than to generate a JPEG image and the compression ratios that give satisfactory image fidelity are only a bit better that in common JPEG. For this reason, we only recommend JPEG2000 on the server side but not as an exchange format in WMS.

A WMTS server has to specify a list of scales where the image can be served and the tile size and origin of each scale. This will define a pyramid of resolutions (see figure 5) that can be obtained by some interpolation technique like a Gaussian pyramid (Burt 1983). In a JPEG2000 based WMTS server the WMTS zoom levels will correspond to the resolution levels of the JPEG2000 internal file avoiding any need of resampling and interpolating data. In the service metadata document, a JPEG2000 based WMTS server will advertise the base resolution and powers of two of this basic resolution in a list of scales that will coincide with the resolution levels internally stored in the JPEG2000 file.

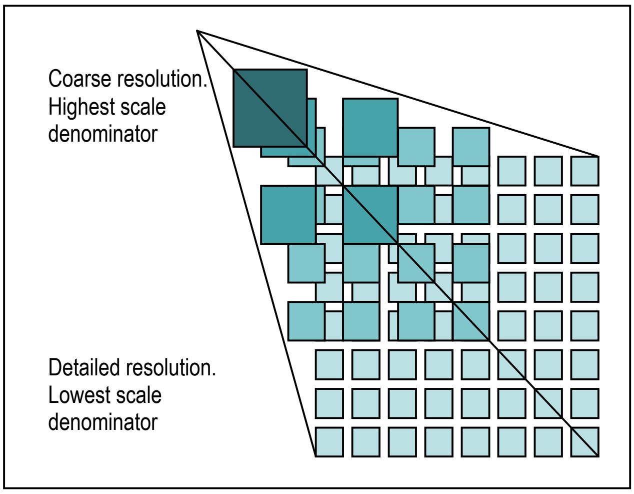

Figure 5. Scale levels and tile matrices in WMTS

WMTS client requests only a regular piece of information from a particular scale level using the tile column and the tile row indices of the tile matrix defined for this scale (see figure 6). Fortunately, JPEG2000 has a similar concept that permits to divide the image in tiles (see figure 7). Nevertheless, there are differences between the JPEG2000 tile model and the WMTS tile matrix set

The extraction of a small region of a JPEG2000 file is a complex process that can take time because the organization of the file is based on codeblock in the wavelet transformed space, so that a small portion of the original image can be distributed in many codeblocks spread around the JPEG2000 codestream. Internally dividing the image in JPEG2000 tiles solves this problem because each tile is compressed independently and is stored in an independent part of the JPEG2000 codestream (Gormish 2003). To make the extraction of a tile easier, the JPEG2000 image stored on a server site should have the same tile structure of the more detailed resolution scale (lowest scale denominator) of the WMTS layer.

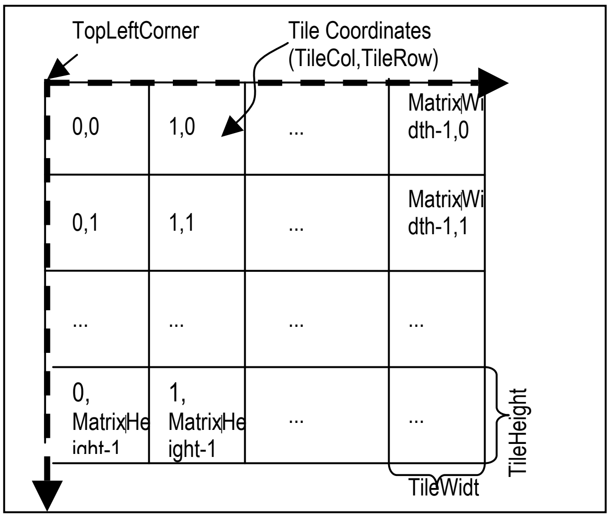

Figure 6. TileMatrix in WMTS scale level

The main difference between the WMTS model and the JPEG2000 model is that WMTS defines the tile matrix of each scale independently, but JPEG2000 defines only a tile structure for the base resolution level. For that reason, a JPEG2000 based WMTS server has to define the next scale (with scale denominator double of the previous one) with the same origin (TopLeftCorner) and the same tile width and height in pixels (that results in a tile that covers a region double of the previous scale level). Annex E of WMTS standard already provides some well known scale sets that follow this rule.

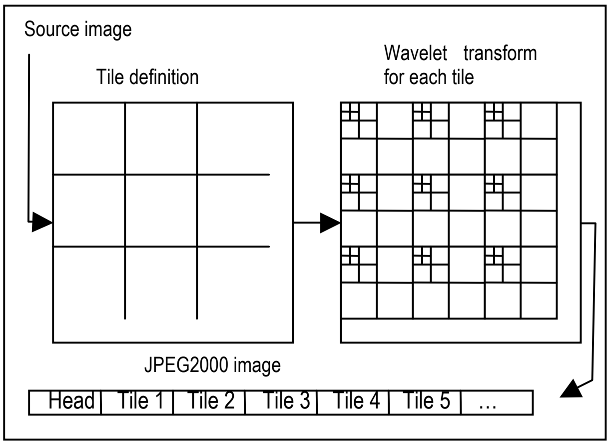

Figure 7. Tiles in a JPEG2000

Following this schema, the extraction of a base scale level WMTS tile will involve one JPEG2000 tile that needs data from all internal JPEG2000 resolution levels. The extraction of a tile of the next WMTS scale level will require reading data from 4 JPEG2000 tiles but accessing codeblocks from all JPEG2000 resolution levels except the base one. The extraction of a tile of the following scale level will require reading data from 16 JPEG2000 tiles but accessing data from all JPEG2000 codeblocks of all resolution levels except the 2 more detailed, and so on.

A WMTS server could offer common formats like PNG or JPEG, but it can also provide JPEG2000 format. In this case, the response image does not need to be uncompressed and recompressed to be served. A base resolution level WMTS tile can be obtained extracting the JPEG2000 tile codestream and generating a new header for this single tile. Other resolution levels will require extracting several JPEG2000 tile codestreams and transcoding them to a coarse resolution (eliminating the more detailed resolution levels).

In conclusion, the JPEG2000 file, that will be used in a WMTS, will perform much better if it is prepared (compressed) with the small tiles typically used in a WMTS service (*i.e.*, 256x265 or 512x512) and with all the resolution levels needed (in general, more than the common default parameters of the compressing tools). Then, this structure is exposed in the layer description of the WMTS service metadata document as a base scale level and the next resolution levels will share the same origin and sizes up to the number of resolution levels on the JPEG2000. Then a WMTS server that receives a WMTS request will have to translate it to JPEG2000 decompression routine commands (see figure 8). If we choose the Kakadu decompressor (considered one of the fastest available) we will need to translate the WMTS parameters to Kakadu syntax (-reduce <discardLevels> and -region {<top>,<left>}, {<height>,<width>}).

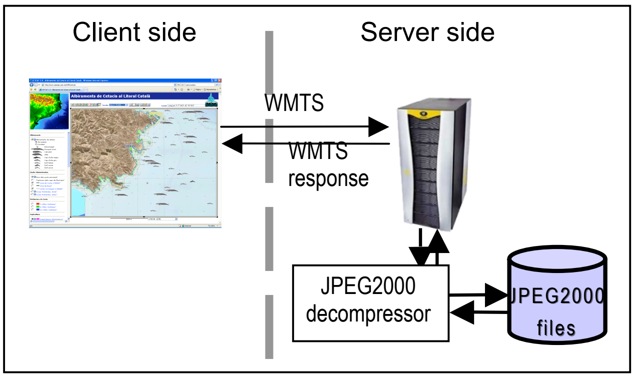

Figure 8. WMTS based on JPEG2000 images on the server

The following expressions translate between WMTS request indices to the Kakadu decompressor parameters:

$$
\begin{aligned}
top &= \frac{tileHeight}{imageHeight} 2^{scaleLevel}\, tileRow \\
left &= \frac{tileWidth}{imageWidth} 2^{scaleLevel}\, tileCol \\
height &= \frac{tileWitdh}{imageWitdh} 2^{scaleLevel} \\
width &= \frac{tileWidth}{imageWidth} 2^{scaleLevel} \\
discartLevels &= scaleLevel
\end{aligned}
\qquad (1)
$$

where *top left height* and *width* = Parameters of the Kakadu decompressor syntax that are real numbers within the interval from 0 to 1 (*top+heigh* and *left+width* must not exceed 1). When truncated, *top* and *left* must be up-rounded and *height* and *width* down-rounded.
TileHeight and *TileWidth* = the size of the tile in pixels
imageHeight and *imageWidth* = Size of the complete decompressed *image* in pixels
scaleLevel = A number that represents the scale level: 0 for the base scale level, 1 for the next scale level, 2 for the next scale level, etc.
tileRow and *tileCol* = WMTS tile indices.

There are 2 main differences between WMTS and JPIP. The first relays on the fact that WMTS typically uses common formats (PNG, JPEG) but can eventually use JPEG2000; JPIP uses JPEG2000 only. WMTS can be used in a web browser without any extra software and this makes it ideal for combining it with other web applications in a RESTful way. JPIP is a more complicated protocol that requires a thick client to work. The second difference is that JPIP protocol allows maintaining codeblock cache on client and server, so that codeblocks received in previous transmissions do not have to be transmitted again. This property saves time and bandwidth when requesting a different zoom level from a region of the image that was previously requested with another resolution. WMTS cannot do that and, in the same circumstances, transfers redundant information. Nevertheless, WMTS is designed to take advantage of the web caching mechanism that benefits all the WMTS users of a particular server at the same time. This makes WMTS more appropriate that JPIP for popular web map services.

In the near future, the WMTS standard could be expanded to eliminate some redundant transmissions if a WMTS client has previously received a tile image of the same region of coarser resolution. A possible technique will consist in transmitting the coarsest resolution tile and then only the differential information needed to enrich this tile to the required resolution level. This can be achieved by the Laplacian pyramid described by Burt *et al.* (Burt 1983). However, this solution requires a thick client to combine the information and will not have the extra JPIP protocol advantages in quality scalability, resolution scalability, and flexible spatial random access (Patel 2007).

2.4 GMLJP2

JPEG2000 file is an image file that can be used to contain geospatial information. Geospatial information requires a minimum set of metadata that cannot be directly encoded in a JPEG2000 header, like the bounding box in a coordinate reference system. The GMLJP2 standard specifies the use of the Geography Markup Language (GML) within the XML boxes in the JPEG2000 data format. It establishes the roles of GML in JPEG2000 to address the additional encoding of metadata, features, annotations, styles, coordinate reference systems, and units of measure (Kyle 2006). It is recommended that any geospatial JPEG2000 uses this extension of the JPEG2000 standard to include at least the bounding box, the coordinate reference system and a reference to an ISO19139 metadata document. This recommendation affects the WCS and SOS standards considered before. A useful characteristic of GMLJP2 is the ability to embed vector annotations in a JPEG2000 file. This characteristic could be used to add to the image some additional features on it to make some landmarks more visible or to include textual descriptions. A simple JPEG2000 viewer will not show them but will be still able to show the image. This could be helpful in a WMS or WMTS client GMLJP2 enabled.

2.5 Conclusion

JPEG2000 can be adapted to several OGC web services. Several strategies can be used to combine JPEG2000 or JPIP with WCS or SOS. A WMTS server, that uses JPEG2000 images as an internal format, needs to prepare the JPEG2000 internal images using JPEG2000 tiles. WMTS scale sets and tile matrices have to be restricted using an small set of rules that will make the service faster and will allow the server to respond in common formats or in JPEG2000 formats easily.

2.6 References

de la Beaujardiere, J., 2004 , OGC Web Map Service (WMS) Interface, Ver.1.3.0, OGC 03-109r1.

Burt PJ, EH Adelson, 1983, The Laplacian Pyramid as a Compact Image Code. IEEE Transactions on Communications, Vol. Com-3l, No. 4.

Christopoulos C., Skodras A., Ebrahimi T., (2002) "TheJPEG2000 still image coding system: an overview", IEEE Transactions on Consumer Electronics, Vol. 46, No. 4, pp. 1103-1127

Di, L., G. Yu, N. Chen, M. Min, H. Wang, 2007, Support of Asynchrony in Sensor Web. Center for Spatial Information Science and Systems 2007.

Gerlek M.P., 2006, OWS-4 IPR for WCS Support for JPEG 2000. OGC 06-128.

Giacovelli P., M. P. Gerlek, 2007, WCS 1.1 Application Profile for JPEG 2000 Coverage Encoding, version 1.0. OGC 07-145.

Gormish M., S. Banerjee, 2003, Tile-Based Transport of JPEG 2000 Images. N. Garcıa, J.M. Martınez, L. Salgado (Eds.): VLBV 2003, LNCS 2849, pp. 217–224.

ISO/IEC 15444-1, 2000, JPEG2000 image coding system - Part 1: Core coding system.

ISO/IEC 15444-9, 2003 JPEG 2000 image coding system – Part 9: Interactivity tools, APIs and protocols

Keens S., 2008, OWS-5 WCS JPIP Sub-setting Engineering Report. OGC 07-169.

Kyle M, D Burggraf, S Forde, R Lake, 2006 GML in JPEG 2000 for Geographic Imagery (GMLJP2) Encoding Specification. OGC 05-047r3

Maso J., Pomakis K., Julia N., 2005, OGC Candidate Web Map Tile Service (WMTS) Implementation Standard, Ver 0.6.0, OGC 07-057r6.

Na A, Priest M., 2007, OGC Sensor Observation Service SOS, Ver. 1.0.0 OGC 06-009r6

Patel S, S Yuval, D Lau , 2007, Interactive Image Browsing with JPEG2000,JPIP. Master project in Image Communication I. Stanford University. , URL: http://www.stanford.edu/class/ee398a/projects/reports/LauPatel Yuval.pdf (accessed 26 Nov. 2009)

Rajabifard, A., Mansourian, A., Valadan Zoej, M.J. and Williamson, I.P., 2004, Developing Spatial Data Infrastructure to Facilitate Disaster Management, Proceedings of GEOMATICS'83 Conference, 9-12 May, Tehran, Iran.

Taubman D.S., Marcellin M.W, 2002, JPEG2000: Image compression fundamentals, standards and practice. Kluwer, Academic Publishers.

Whiteside A, J.D. Evans, 2008, OGC Web Coverage Service (WCS) Implementation Standard, Ver. 1.1.2, OGC 07-067r5.