magHD: a new approach to multi-dimensional data storage, analysis, display and exploitation

Christophe Angleraud
Chief Technology and R&D Officer, Magellium, Toulouse, France
E-mail: christophe.angleraud@magellium.fr

Abstract. The ever increasing amount of data and processing capabilities – following the well-known Moore’s law – is challenging the way scientists and engineers are currently exploiting large datasets. The scientific visualization tools, although quite powerful, are often too generic and provide abstract views of phenomena, thus preventing cross disciplines fertilization. On the other end, Geographic information Systems allow nice and visually appealing maps to be built but they often get very confused as more layers are added. Moreover, the introduction of time as a fourth analysis dimension to allow analysis of time dependent phenomena such as meteorological or climate models, is encouraging real-time data exploration techniques that allow spatial-temporal points of interests to be detected by integration of moving images by the human brain. Magellium is involved in high performance image processing chains for satellite image processing as well as scientific signal analysis and geographic information management since its creation (2003). We believe that recent work on big data, GPU and peer-to-peer collaborative processing can open a new breakthrough in data analysis and display that will serve many new applications in collaborative scientific computing, environment mapping and understanding. The magHD (for Magellium Hyper-Dimension) project aims at developing software solutions that will bring highly interactive tools for complex datasets analysis and exploration commodity hardware, targeting small to medium scale clusters with expansion capabilities to large cloud based clusters.

1. Introduction

As the capacity of data storage & processing increases, the spatial and spectral resolution of raster or vector data also increases, as well as the complexity of the phenomena that the scientists or the engineers attempt to explain. Strangely enough, the media used to explore and display these ever more complex data landscapes did not evolve that much in the last decades: keyboard, mice and colour screens.

Although screens are bigger, flatter and more colourful, mice are thinner and more reactive; we must admit that since Google™ Earth has brought a brilliant dynamic GIS visualization environment on everyone’s desktop, no important new technique have come around to help understanding easier and deeper the vast amounts of data available.

Recently though, so called Cloud Computing has quickly turned computer farms into large arrays of hard disks, allowing peta-bytes of data to be persisted and processed. However, low cost and long lasting persistence is now a reality but, to our knowledge, no software is yet taking advantage of these
new environments for providing efficient data access, visualization and analysis on commodity hardware, as far as geospatial information is concerned.

magHD is attempting to make use of high performance computing frameworks for hyper-dimensional datasets, using the best of the breed distributed storage, distributed processing as well as volume data exploration strategies used in medical imaging in order to create a highly interactive visualization environment that will facilitate understanding of complex phenomena.

We also believe that real-time interaction with end-users is one of the keys to achieve this goal by using eye persistence unconscious abstraction performed the brain. magHD will also provide new ways of interacting with the data, for example by using smartphones and tablets to provide simultaneous collaborative displays, yet preserving individual explorations and analysis.

The applications of magHD could be potentially quite large both in the civil and military domains: scientific visualization, climate analysis, GIS science, smart augmented reality GPS, complex environmental analysis, assessment and picturing (Recognized Environmental Picture & Rapid Environment Assessment), command & control…

1.1 Goals and organization of this document
This document has been written to provide a common view of the magHD project to its participants as well as to seek for potential early adopters and technical or financing partners.
It is organized in 4 sections:
1. Data storage and indexing
2. Data access and processing
3. Interactive data visualization
4. Current status & foreseen use cases

1.2 Previous work
Our work is mainly built upon existing technologies and similar approaches used to explore complex datasets used mainly in medical and scientific imaging. Therefore we do not claim a highly innovative scientific work but rather try to transpose these existing techniques into a new area of use: GIS and earth sciences where vast amount of geospatial data have to be explored and combined together into a consistent exploitation and visualization environments

In the field of computer science, many new techniques have emerged these last 10 years with regards to “big data” and “cloud” solutions. Among those, we would like to cite here the efforts of the Apache Foundation with its Hadoop [1] framework. This framework is quite comprehensive and offers a large collection of technologies that can be used in the building of our multi-dimensional data analysis, exploitation and visualization environment.

Despite the existence of such large framework, we consider that it generally lacks some native capacity to properly index and distribute the kind of data that environmental analysis requires. Namely, we believe that an essential part still missing is a proper multi-dimensional indexing and distributed storage solution. Some technologies are providing entry level spatial indexing solutions, such as mongoDB [2], Lucene Spatial API [3], or Spatial4J [4] for Solr [5], but these entry level are focused solely on the 2D indexing (longitude, latitude) case for mapping applications. Some work has also been done on using Hadoop and/mapreduce style of parallel processing in order to perform 2D spatial operations: [6], SMJR [7] and Pigeon [8].
In order to build a more generic solution, that could be used with general purpose multi-dimensional datasets (such as meteorological forecast fields for temperature, wind speed that would require simultaneous access to and query of X, Y, Z, T and vector value fields), one would need an embedded distributed multi-dimensional indexing scheme. Some interesting extensions to the previous distributed 2D spatial indexing work have been published: [9] and [10].

Therefore, we can see that “big data” or “cloud” infrastructures are on the verge to provide efficient scalable solutions for multi-dimensional data. The next question is raised by the amount of data and the large differences of data types that scientists and end users will have to cope with on a single (or dual) screen environment. We believe that to provide efficient analysis and display environments it is necessary to provide highly interactive processing capabilities to provide new ways of combining complex data. These kinds of approaches have been used in the medical imaging area, in order to ease the exploration of volumetric datasets. We can cite here the techniques called “importance driven visualization” [11].

We also would like to cite existing software packages that offer interesting features that show partial feasibility of our approach: Paraview [12] and Makai Voyager [13].

2. Distributed multi-dimensional data store
In order to provide a highly interactive environment able to cope with very large multi-dimensional datasets, the first challenge to tackle is proper indexing and storage.

Recent progresses have been made lately on big datastores through distributed file systems (DFS) as well as noSQL (Not Only SQL) databases. These new technologies are now reaching a first degree of maturity, proving sustainable performances and reliability but the applications are dedicated to textual processing (web pages and social networking indexing).

magHD will make use of the best available software techniques such as distributed RAM cache, distributed databases and distributed disk storage using multi-dimensional on-the-fly indexing to provide efficient data search and retrieval. We do not intend to re-invent the wheel but, to our knowledge, no cloud environment provides yet efficient techniques for building distributed multi-dimensional indexes.

We aim at using the most efficient spatial indexing techniques, such as R-tree, kd-tree, Gist, in a fully distributed cluster to allow for parallel search and extraction in a peer-to-peer kind of communication scheme.

We also believe that the rather old approach to data modelling used in GIS and relational databases is a strong limit to data re-use and sharing. Therefore, our approach will be to favor tag like attributes and to favor multiple overlapping models to hold the information. This dynamic non-restrictive approach will make use of dynamic filtering and combination (using the map-reduce paradigm) in order to extract semi-structured information in destination to the end user applications.

This approach should also be quite appropriate for handling different representations of common objects. For example, the set of attributes for a given road could show common values (name, id, number of lanes…) as well as distinct ones according to the end users application (traffic information, pavement type, administrative authority…). This arrangement bed on tag sets belonging to different namespaces will also allow multiple views matching different normative requirements to be extracted from the same data set. In a similar way, the geometry of the road itself could be multiple: central axis for GIS mapping applications and separate lanes and kerb for road construction/maintenance applications. The geometries could also handle different level of details to be hold together.
Given this requirement of flexibility, we believe that the distributed key-value stores or the document oriented noSQL databases are a good fit, provided, multi-dimensional indexing is added (some solutions such as mongoDB or memBase are already providing simple 2-dimensions indexes, we want to broaden this to any number of dimensions).

This native multi-dimensional indexing will allow magHD to cope with a wide range of applications requiring different types of data, such as geosciences, meteorology/climatology, oceanography, hydrography, geography… The proposed approach will mix Big Data storage issues with native indexing and distribution to provide the best performing solutions for fast data search and access.

3. Data Access and processing
The coupling of data storage issues with native indexing and distribution over machine clusters represent the main building block of efficient access and processing.

Our idea is to promote distributed processing strategies on machine clusters that favour the execution of processing on the very same node where storage happens. This strategy is not new; it has been vastly used on large clusters for web 2.0 applications as well as through the Map/Reduce paradigm promoted first by Google and then by the Hadoop open source cloud framework.

The extension of this paradigm to data processing dedicated to multi-dimensional data sets that we want to achieve within magHD is based on the direct use of the native multi-dimensional index to identify the storage nodes holding data intersecting the area of interest of the processing request.

The processing requests will therefore be split and distributed on all the nodes holding intersecting data. Thus we intend to extend our Ingrid data processing framework to make its workflow engine able to run into a fully distributed environment.

Specific attention will also be given to data blocks storage and transfer. It appears to us that JPEG2000 is a very versatile and efficient storage format and could be very efficiently used in such distributed storage environments (for earth observation images but as well for a lot of earth science gridded data sets). Our intent regarding JPEG2000 is to forget the file storage structure and to make direct use of the wavelet packets entities to spread them across the distributed data store.

One of the main asset of our processing framework will be its ability to discretize the processing requests into gridded processing chunks (the Map phase of Map/Reduce) so that the results of overlapping processing requests could be stored into an interim distributed cache. For example, a processing request that requires belongs to a real-time display cession, will always be split with exactly the same discretized bounding boxes and resolutions to maximize the chances that the processing results will be reused by the next partially overlapping visualization request. Therefore, the definition of these discretization grids need to be shared among all the processing nodes and exploitation work-stations, this will be part of the processing chains metadata.

Our main focus being interactive visualization, we want to take advantage of the pauses occurring during the human manipulation of the software, when the end user is performing analysis tasks. This means that during an interactive cession, we have bursts of data I/Os and CPU during view motion but then the CPU or GPU resources are much less solicited and could therefore be lent to other users.

Consequently, we envision a collaborative environment where each node could, in turn, play the role of a server or a client of other nodes, at any time. Thus, the available resources would therefore be used at their maximum potential at the global cluster level.
As a wrap up of the previous section, we provide in Figure 1 an overview of the different components that magHD is being built upon.

![Figure 1. magHD Architecture overview.](image)

4. Interactive Data Visualization

More data makes users more confused. The vast majority of current data exploitation and exploration software propose layered data organizations, each layer being independently rendered through style sheets on the screen. The GIS software proposes to filter out some data based on the displayed scale in order to reduce screen clutter.

We believe that it is possible to make better use of the current graphics engine by using much more interactive techniques.

4.1 Real time interaction

First, adopting fast rendering GPU-aided techniques can open new fields of interaction with the end-user, thus, allowing psycho-visual perception to increase the amount of information exposed to the user’s brain.

For example, using rapid cursor manipulation to dynamically explore time or to change particles density or contrast of images can help to understand phenomena if, and only if, the users gestures are immediately changing the displayed data, in real-time. The unconscious associations of the gesture with what the eyes are perceiving allow temporal integration of complex changes by the brain.

Particle animation techniques could also be used to show data cube contents without occluding the background information. Again, to allow for psycho-visual integration, the particles have to be displayed very smoothly. This is possible thanks to GPU programming techniques mainly used in video-games called shaders (sort of GPU hosted highly parallelized micro-software).
Figure 2. Particles used to show a cubic wind field while keeping the earth background visible.

Figure 3. Dynamic interactive plane clipping attached to camera motion.

Second, we believe that allowing real-time navigation through complex datasets is a way to allow intuitive understanding of the displayed contents. For example, the forward-backward motion of the camera can be used to move a temperature clipping plane in order to show local variations that can reveal specific phenomena to take into account.

4.2 Importance driven appearance

In order to reduce the clutter on screen, we believe that the data appearance should be able to change by itself, according to the other kind of information displayed at the same screen location. Our goal is to make data more intelligent so that its appearance could change in presence of other overlapping data.

Figure 4. Importance driven building drills a transparency hole through a temperature cube.

Figure 5. In this case we requested that the current field was more important than the temperature field.

This concept is borrowed from medical volume visualization techniques, called “importance-driven visualization”. This technique aims at automatically changing the transparency of voxels occluding
more important voxels that could be hidden. We propose to adopt the same principles and to extend it to other properties than transparency.

Namely, we envision a new way of specifying data rendering: each data record displayed on the screen could embed invisible importance information (that would be derived from the requirements of the end users). This importance information would then be interpreted by the graphic engine during the rendering passes in order to adapt the appearance of less important objects. For example, it’s the transparency, blurriness, color saturation, filling / wireframe, line width, etc. could be modified dynamically in order to reveal the important features but keeping less important information slightly visible to deliver the overall contextual (more global) information.

For example, as shown on figure 3&4 controlling transparency can help to keep the most valuable information in front of the eye even though it is located behind other, less important information that would normally be occluding the more important one. We could make use of this feature for new generation GPS devices that would make use of collaborative distributed vehicle locations (like AIS for ships) in order to show potentially colliding vehicles occulted by buildings or any large obstacle on the screen of a smart augmented reality3D GPS.

5. Current status & foreseen use cases
At this stage, we have developed a prototype implementation which shows the feasibility of the magHD concepts. Currently, the developments are mainly dedicated to the visualization engine. We have a first version of the magHD SDK running and aim at being able to release a first operational version of this SDK by the end of 2013, as some illustrations of this paper show.

Meanwhile, we have also started some experimentation with distributed storage and processing, buts this area will still need some work, especially concerning multi-dimensional distributed indexing and proper use of distributed caches.

We envision multiple applications for this technology: scientific visualization & general public showcases, oil & gas sub-sea and sun-surface visualization of complex data sets, climate change monitoring, defense REP (Recognized Environmental Picture) & REA (Rapid Environment Assessment) applications, defense and civil command & control, lightweight embedded 3D guiding (augmented reality GPS) and we are currently seeking partnerships or early adopters in order to further investigate advanced visualization and analysis techniques and to build actual applications.

6. Conclusion
magHD has now proven the feasibility of most of its challenging features and is now ready to enter into a new R&D phase dedicated to large distributed storage & processing and collaborative work. Version 1 of its Hyper-Dimension data visualization engine will be ready by the end of this year. It will be package as an SDK ready to be used for developing new innovative smart data-centric applications.
7. References

[1] Apache, "Hadoop," [Online]. Available: http://hadoop.apache.org.

[2] MongoDB, Inc., "mongoDB," mongoDB, [Online]. Available: http://www.mongodb.org.

[3] Apache Foundation, "Lucene Spatial API," Apache, [Online]. Available: http://lucene.apache.org/core/4_0_0/spatial/.

[4] Apache Foundation, "Spatial4J," Apache, [Online]. Available: https://github.com/spatial4j/spatial4j.

[5] Apache Foundation, "Solr," Apache, [Online]. Available: http://lucene.apache.org/solr/.

[6] E. Tanin, A. Harwood and H. Samet, "Using a distributed quadtree index in peer-to-peer networks," VLDB J., vol. 16, no. 2, p. 165–178, 2007.

[7] S. Zhang, J. Han, Z. Liu and K. Wang, "SIMR: Parallelizing spatial join with MapReduce on clusters," in CLUSTER'09. IEEE, 2009.

[8] A. Eldawy and M. F. Mokbel, "Pigeon: A Spatial MapReduce Language," in ICDE14, 2014.

[9] C. Du Mouza, W. Litwin et P. Rigaux, «A Framework for Distributed Spatial Indexing in Shared-Nothing Architectures, » BDA, 2007.

[10] C. Zhang, F. Li and J. Jestes, "Efficient Processing of k Nearest Neighbor Joins using MapReduce," in EDBT, 2012.

[11] I. Viola, A. Kanitsar and M. E. Groller, "Importance-driven feature enhancement in volume visualization," Visualization and Computer Graphics, IEEE Transactions, vol. 11, no. 4, pp. 408-418, 2005.

[12] Kitware, "Paraview," [Online]. Available: http://www.paraview.org/.

[13] Makai Ocean Engineering, "Makai Voyager," [Online]. Available: http://voyager.makai.com.