# Freedom in geoinformation science and software development: A GRASS GIS contribution

Helena Mitasova*, Markus Neteler**

* Department of Marine, Earth and Atmospheric Sciences, North Carolina State University, Raleigh, U.S.A.
e-mail: hmitaso@unity.ncsu.edu
** ITC-irst, Interactive Sensory Systems Division, Trento, Italy
e-mail: neteler@itc.it

## 1 Introduction

Over the past decade geoinformation field has evolved from a highly specialized niche to a technology with broad impact on society and its interaction with nature. Geographic Information Systems (GIS) applications now range from simple navigation to critical and extremely complex tasks, such as prediction and management of natural disasters. Due to the increased use of GPS, faster access to georeferenced data, expanding field of remote sensing and real-time monitoring, GIS technology is entering many new disciplines and industries and GIS is becoming a part of general computational infrastructure. It is therefore natural that geospatial tools are being developed also within the Open Source and Free Software community [60].

Several important trends can be identified in the current development of geospatial technology:

- modern, highly automated mapping and monitoring technologies produce massive amounts of spatio-temporal georeferenced data (e.g., LIDAR [39], [53], IFSARE, 3D digital photogrammetry, multi-spectral imagery, real time kinematic surveys, interferometric side-scan sonar, remote sensing data etc.);
- on-line distribution of data shortens the time between the data acquisition and data distribution, and supports near-real-time data availability (e.g., USGS Seamless data sets: NED and NLCD [31], NOAA LIDAR data retrieval tool LDART [17], USGS Real-time Water Data [32]);
- growth in web services [20] is reflected by dramatically improved WebGIS sites that are adding tools for data processing (e.g. on-fly projections) and spatial analysis to its most common data viewing and navigation capabilities [47], [54];
- parallel and distributed computing (GRID) technologies are being implemented for processing of large-scale GIS data sets (especially for remote sensing and terrain modeling, [56]);
- location based services, wireless spatial information brokerage are the main focus of commercial geospatial technology development, but Open Source initiatives are emerging too (e.g. [62]);
- Open Source and Free Software is being increasingly used in mainstream systems [11], [14];
- improving interoperability and standardization is reflected in adaption of GML [18], GeoVRML [12] and other standards.

The software components of geoinformation technology have a profound impact on the capabilities to effectively use the spatial data for solving a particular problem. To ensure continuous innovation and improvement, existence of diverse approaches to GIS software development is crucial. Besides the widely used proprietary systems, an Open Source and Free Software GIS plays an important role in adaptation of GIS technology by stimulating new experimental approaches and by providing access to GIS for the users who cannot or do not want to use proprietary products.

GRASS, as an Open Source/Free software, is well positioned for the current geospatial technology environment, however, the modernization of its more than 10 years old core code is needed as will be discussed later.

It appears that GRASS has successfully survived many years without an official institutional or industry support and found its place in the Open Source and Free Software community. It has a critical number of developers and users, growing support infrastructure (distribution sites, mailing lists, literature, courses, national user meetings), and now the first international conference, since it was released under GNU GPL. In the next sections we discuss the ideas that have stimulated GRASS evolution, assess the current development and management model and present our views of GRASS future.

## 2 GRASS GIS evolution

GRASS was developed in 1982 - 1995 by the U.S. Army Corps of Engineers Construction Engineering Research Laboratory (CERL) in Champaign, Illinois, to support land management at military installations. Since late $80^{ies}$ the coordination of GRASS development was handled by the GRASS Inter-Agency Steering Committee (GIASC), a non-profit corporation. It has evolved into a more independent Open GRASS Foundation to enable technology transfer of GRASS while keeping it open [25]. The foundation later re-focused on interoperability, adopted a new name Open GIS Foundation and became a Consortium (OGR) [26], [8]. CERL ceased its official sponsorship of GRASS and in the following transition period GRASS lost most of its support and users, who moved to off-the-shelf proprietary technology.

After forming a new development team, the turning point in the recent GRASS history was the adoption of GNU GPL (General Public License, see `http://www.gnu.org`) in 1999. By this GRASS embraces the "Free Software philosophy" [30], [29], well known from the GNU/Linux development model, which stimulated its wide acceptance ([21] and [22], for a discussion see also [34], [60]). GRASS has also a long history of Internet distribution, starting with GRASS 4.0 release in 1991, which made it available to users all over the world. Internet continues to play a major role in the GRASS development coordination and support (Figure 1).

While a lot of GIS software development has evolved around the need to automate the map drawing process, development of GRASS focused from the very start on spatial analysis needed for land management applications. The development followed directions or "visions" which are, to certain extent, still valid, as it is reflected by the current developments.

**GRASS as easy to use, cost effective land management tool** was designed with raster data processing at its core, further enhanced by comprehensive GUI-XGRASS and vector capabilities aimed at supporting cartographic output. Raster based spatial analysis has continued to expand, and its applications are well illustrated in the proceedings (e.g., [36], [59] [46], [57], [52]). Enhanced vector capabilities (2D/3D geometry, multiple attributes with RDBMS support) are being developed for GRASS5.1 [38]. On the other hand

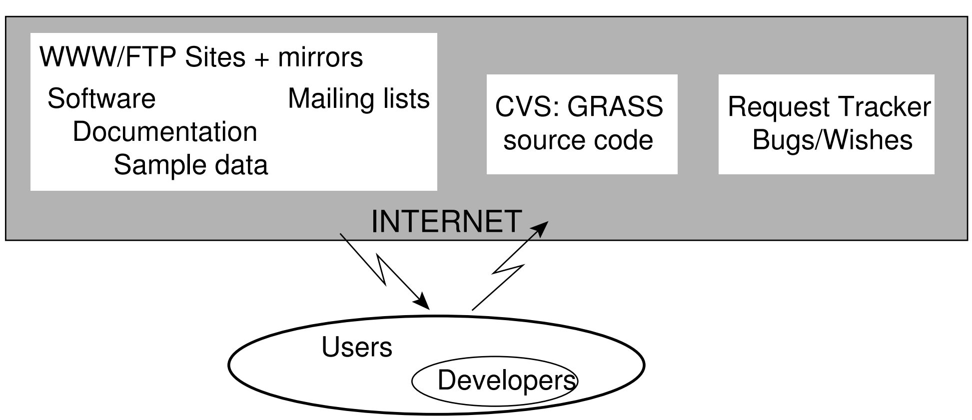


Figure 1: GRASS development model.

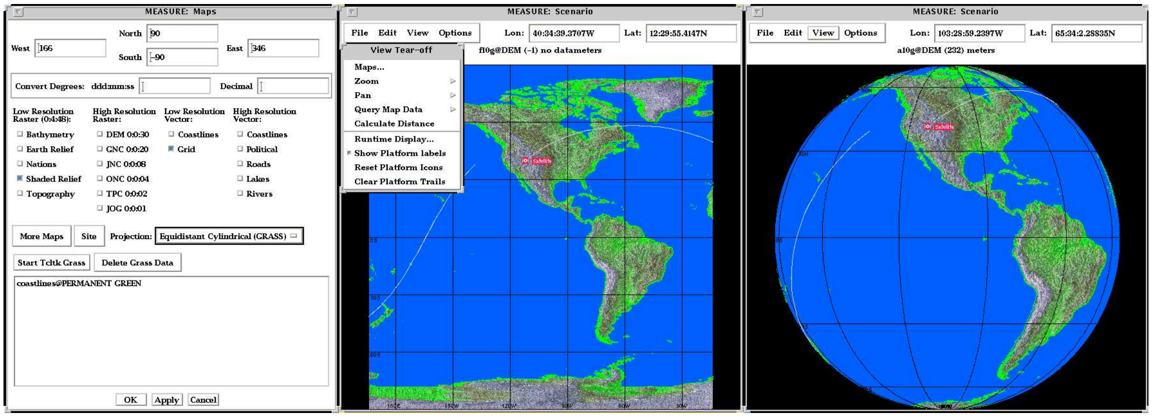


Figure 2: Snapshots of an upcoming graphical user interface for GRASS (developed by B. Wallace, LMCO).

XGRASS is not used any more and some efforts are under way to replace the currently used *tcltkgrass* with more effective GUI (an example is shown in Figure 2). However, cartographic output and viewing maps in general (including legends and scales) still lack the convenience and efficiency of the current state-of-the-art products.

**GRASS as a research environment** has served for exploration of new algorithms, approaches and applications. Land management related research supported significant GRASS development with new modules being added as by-product of these projects. Pioneering work has been done in a number of areas, for example:

- GRASS served as a testbed for coupling environmental and physical process models with GIS (e.g., ANSWERS [23], AGNPS [28], SWAT [27], CASC2D [13], WaterFea [33]). Most of these models are no more supported in GRASS and are linked to proprietary GIS. However, recent developments indicate revived interest in this area, in particular, the implementation of *r.topmodel* [1], *r.sim.water* & *r.sim.sediment* [48], re-design of *r.hydro.casc2d* [55], as well as new models such as evaporation over tilted slopes [41], or solar radiation *r.sun* [49];

- Visualization and 3D dynamic GIS (Figure 3) were developed as a prototype by the team lead by Dave Gerdes [15]. Although it has not been officially released, its development continues, as illustrated by the recent contribution of volume interpolation and analysis module *s.vol.rst* (including option for computation of precipitation with influence of topography) [10]. Another examples are modeling of evaporation processes by *r3.mapcalc* [41] and 3D geology models [51]. Port of *nviz* to OpenGL [3] opened the use of GRASS visualization to non-SGI platforms as reflected for example in papers [65], [53], [40], [51]. Incorporation of existing volume visualization capabilities into *nviz* still waits for a developer.

- Image processing tools have been incorporated thanks to the strong raster data support [35]. After a period of discontinuity new tools are under development such as automated image registration. Further research is described in [45], [63];

- Decision support tools have evolved from the early contributions such as B-infer (Bayesian expert system) [4], [5], to recent applications [46], [50]. Multimedia tools was designed for cultural resources at installations using *xgen* (XCHRIS [7]), however, the development in this area has been limited.

Besides the areas with a long history of development and applications, there is a number of new contributions reflecting the current advances in GIS technology:

- implementation of a new 3D multiattribute vector architecture [38] creates opportunities to extend the GRASS vector capabilities to the level comparable with raster data processing and beyond;

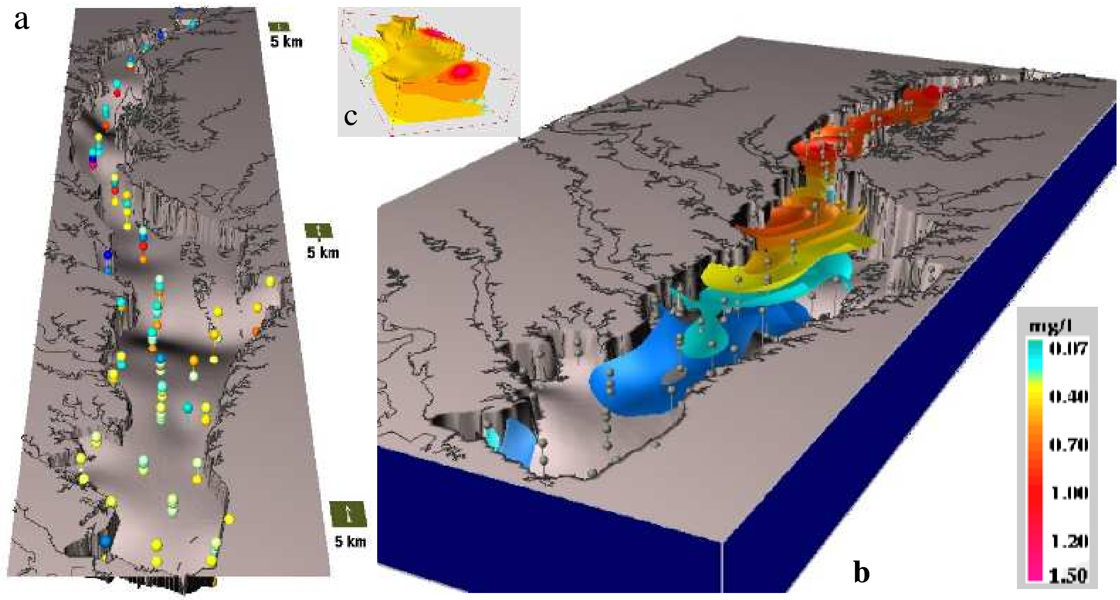


Figure 3: Application of GRASS prototype 3D dynamic GIS from 1993 for study of annual cycle of nitrogen concentrations in the Chesapeake Bay, USA, a) given data points colored according to the predictive error of interpolation estimated by crossvalidation (the RST crossvalidation program is being updated and prepared for release); b) snapshot from animation based on quadvariate interpolation of time series of volumetric data using *s.volt.rst* and visualized by *sg4d*. While trivariate interpolation has been released with GRASS5.0, *s.volt.rst* still waits for some updates. The animation can be viewed at `http://skagit.meas.ncsu.edu/~helena/gmslab/viz/movies/chall4d.gif` c) experimental solid volume model created from the subset of the data using *showdspf*, its successor *r3.showdspf* and G3D library needs some fixes to get the full capabilities of the original program.

- development of “babyGRASS” for wireless and mobile GIS e.g., using iPAQ handheld, opens possibilities to bring GRASS into the field [62], [64], [44];

- several Web mapping applications and a Web interface to GRASS bring GRASS databases and tools online by using a combination of GRASS with other Open Source and Free Software tools, such as PHP, MapServer and PostgreSQL [47], [54], [16];

- functionality of GRASS is being expanded by adding a number of new modules, from wavelet applications [65], solar radiation [49], path sampling for hydrologic and erosion modeling [48], LIDAR site data processing support [39] and many others, as demonstrated by the conference proceedings;

- GRASS is being applied in several new fields (besides its traditional environmental and land management), for example, in geology [51], [50]; coast and sea [61], [53], space [43], atmospheric [41], [42], real time hydrology [55];

- linkages to other Open Source and Free Software projects such as *gstat* or *R* statistical software [2], [37] continue to rapidly expand and strengthen the power of Open Source and Free Software geospatial computing environment.

Increasing number of contributions demonstrates that GRASS is a living system that builds upon its traditional fields in landscape management and environmental modeling and expanding into new areas in response to the developments in the mapping and geospatial technology.

# 3 Current GRASS development model

The GRASS history shows a continuous change in the development model. Started as rather unusual project, the code has always been public, first in public domain, and later released under GPL. From the previous section we can identify several phases of GRASS development (cf. Figure 4). After several years of systematic growth and innovation (from GRASS 1.0 to GRASS 4.2), and a transition phase without centralized code control and development a long consolidation phase followed. During this phase numerous 5.0beta and 5.0pre versions have been published. This consolidation phase has led to a portable, stable and reliable code base along with several feature enhancements. However, to keep up with the most recent GIS developments and user needs, a new phase of innovation is needed. GRASS 5.1 development has been started, but many more active developers are needed to put GRASS at the forefront of GIS again.

The GRASS development model is embedded in the Free Software movement and style and uses well known tools for code management and dissemination (c.f. Figure 1). Since code development is strongly related to the team taking responsibility for the project, a few considerations will follow.

## 3.1 Components of the development and maintenance

The main components of the development and software maintenance are built on top of a highly automated web-based infrastructure. This is sponsored by ITC-irst (Trento, Italy) and numerous other sites worldwide where the code base is mirrored. The most important task is coordination of development which is done through the developers mailing-list. Currently, the GRASS development team consists of 7-15 active developers, while more than 220 subscribers (8/2002) follow ongoing discussions and sometimes contribute. On the other hand, there is a number of developers who work independently from the core team and provide new GRASS modules on their web sites. These modules may be eventually included into releases. The developers are distributed around the globe, beyond political and geographical boundaries, and they focus on stabilizing and improving the code base. The GRASS team can be considered as “council type” as opposed to Raymond’s “bazaar type” and “cathedral type” development methods ([21]). A limited, but open group takes decisions about design and implementation. These developers are also advanced GRASS users who know most of the system’s functionality and view the system also from a user’s standpoint which simplifies communication and improves the user support. The team structure is heterogeneous, there are specialists as well as generalists. Various tools are used for code management and dissemination: *GNU/Linux*, *Apache* web server, *wu-ftp, rsync* mirror software, *MailMan* mailing list software (at time more than 8 GRASS related lists are hosted at ITC-irst), *CVS* (Concurrent Versioning System), GNU *gcc* compiler and further tools.

An interesting question, from a software evolution point of view, is “code ownership”. Since the development team has completely changed after 1995/1997, almost nobody was familiar with the entire code and (undocumented) design issues. The recent history shows that orphan code is regularly adopted, fixed or modified, or eventually completely rewritten. Most developers specialize in certain programming and therefore are able to contribute at a high scientific level. The code coordinator is taking care of quality issues:

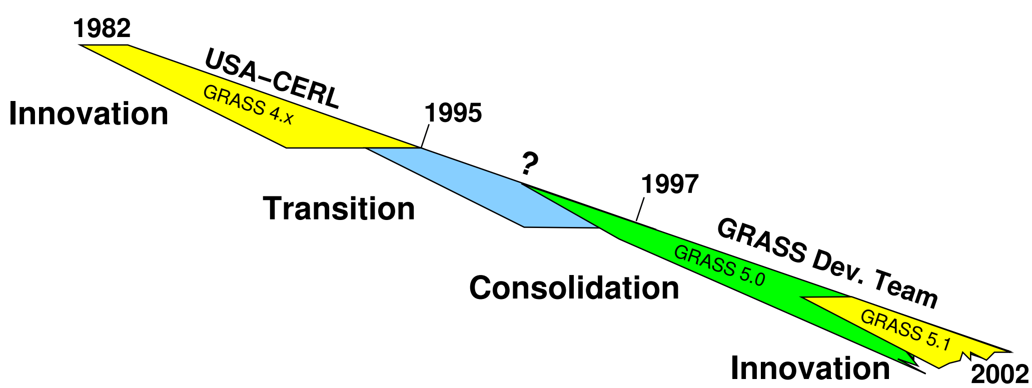


Figure 4: GRASS development phases.

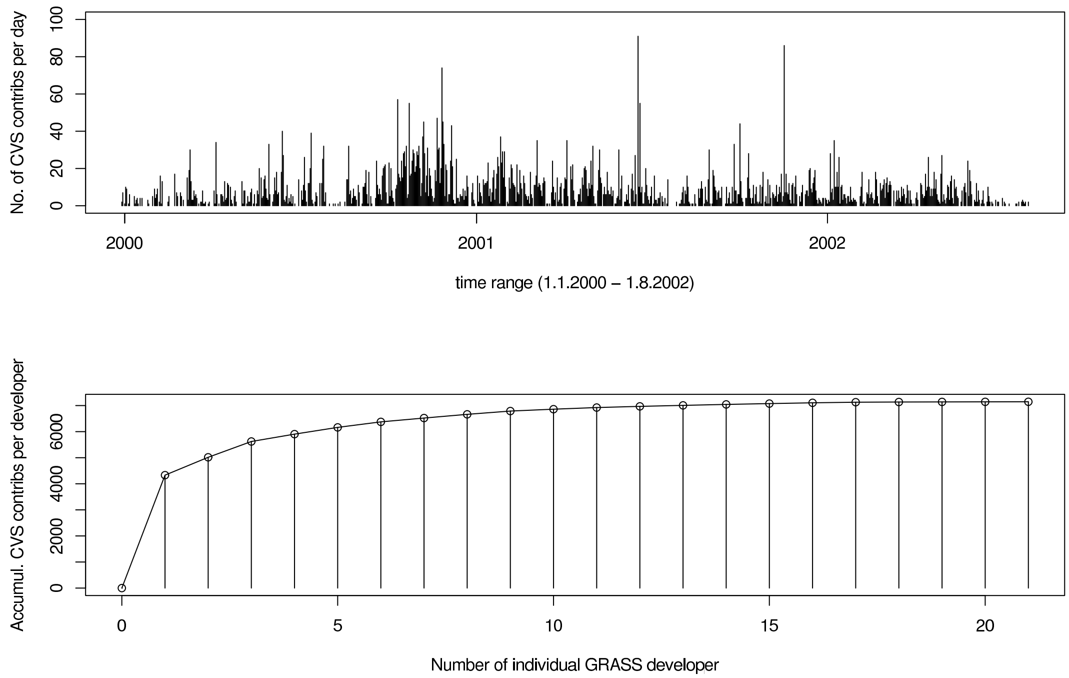


Figure 5: Survey of submissions to GRASS CVS server from January 1st, 2000 to August 1st, 2002 (based on ChangeLog, initial code submission excluded). The upper graph shows the number of daily code submissions. The lower graph represents the accumulated number of code submissions per developer, reaching more than 7000 in total. It must be stated that submissions of developer 1 include estimated 15% submissions for other developers without CVS write access.

he supervises, as much as possible, the submission of portable code (ANSI-C), updates of documentation, he often tests changes, explains the code structure to newcomers, etc. The coordinator plays also a role in "plugging together people" who do not know each other but share interests (e.g. as for the new vector networking library). These shared interests often lead to faster code development and increased satisfaction. It is important to keep in mind that no developer is irreplaceable, however, it is necessary to ensure that developers who are leaving the team have documented their contributed code. Decisions on code submission, substitution or deletion are taken in consensus (usually in a "if-no-objections" mode). A notification mailing list distributes the source code differences (diffs) as sent from the CVS server to several developers for verification. Figure 5 shows the number of code submissions to CVS (daily submissions and accumulated on a per-developer base) for the time span of more than two years since the source code has been managed in a CVS repository.

Several GRASS versions are available, but not every version is intended for every user. Linux kernel numbering scheme of releases is currently being adopted. All *major.even.something* versions are intended to the general user and considered as stable (e.g. 5.0.x), while *major.uneven.something* are development versions (5.1.x). The "response time" for bug reports is, as also known from other Free Software projects very short: often bugs are resolved within one or two days, the updates made available through CVS. A thorough study on this issue may be interesting in terms of software engineering and code evolution, particularly with regard to the enormous time span of GRASS development.

Overall, the emphasis of the GRASS development and management team work is on coordinating two major efforts:

- stable releases for routine users with focus on reliability;

- experimental environment for implementation and testing of new data structures, code organization, new modules etc. ([38], simulations ([48]), mobile GIS ([62]), new algorithms (e.g., [10], [39], [49], [65]).

After release of the stable version 5.0.0 the focus will shift onto 5.1. A new vector architecture overcomes the limitations of GRASS 4.x-5.0 by extending the vector support with attributes stored in external relational databases and by new 3D capabilities [38]. Besides internal file based storage the geometry may alternatively be stored in PostGIS database. This enables users to maintain large data sets with simultaneous write access. External GIS formats such as SHAPE-files may be used directly without necessity of format conversion. It is suggested to fade out the current sites support and handle point data as vector points (to reduce the code maintenance efforts and to improve the attributes/NULL management).

All required raster code will be migrated to 5.1 only after a major overhaul with clone detection (same or similar code existing in several versions) and library clean-up. Also the display system needs to be substantially improved.

In summary, it can be stated that after more than two years the GNU license is continuously attracting new developers. Also the management of GRASS in CVS has stabilized the development and taken unnecessary administrative workload from various shoulders. The other management tools (such as Request Tracker, mailing lists, etc.) support one of the most important issues in Free Software development: communication.

## 3.2 A long wish list from user's and developer's perspective

Several areas can be identified with a strong need for improvement in terms of functionality and code structure.

A perennial question for new GRASS users has always been: "Why not make things easier?" A number of developments should respond to this question, including further simplification of import/export capabilities as well as support for current and upcoming data standards. Not surprisingly, it has been suggested to incorporate libraries such as GDAL and OGR developed by Frank Warmerdam for this task. A more flexible definition of LOCATION may reduce the difficulties for getting started with GRASS and improve management of large projects which involve work with different coordinate systems and regions. Import/export and work in multiple projections can be further simplified by introduction of projection on the fly, with map coordinate system information stored with the individual maps, in a standardized format (EPSG codes etc). This approach is known from MapServer. Unfortunately, the processing time for large raster maps may be too slow for some applications.

An urgent need from a user's perspective is a new display system which supports cartographic elements and easy map printing. Besides the redesign of the basic map viewing tools, academic research may lead to new approaches such as the MIT tactile visualization tools [58]. To really enjoy the local map collection, map management could be enhanced through a new geo-browser. This tool may provide also graphical data import functionality, similar to a file manager. The browser should display for each selected map general information such as color depth, projection information, data size and more.

For various scientific applications, an improved spatio-temporal data management (3D/4D/time support) is needed. While new sensors provide a continuous stream of geo-information, adequate storage and analysis tools are still missing. One of possible solutions to MAPSETS with massive amounts of raster maps representing time series may be storage of 2D time series raster data as a block, e.g. in HDF formatted files or using the GRASS G3D format. HDF also provides support for time series of volumes. Although GRASS provides a DateTime library, it may be replaced by the DateTime library from R as it is portable and heavily tested [24]. However, implementation in terms an analysis support requires a large number of modifications and/or development of new tools, so this would be a major project. Nowadays and especially in future data may arrive in real time from distributed sensors (e.g. environmental monitoring). It should be possible to process these data in real-time: while data are sent through polling into the system (e.g. PostGIS), GRASS could listen on the other side and handle postprocessing and mapping. User could generate snapshots, timelines, profiles etc in all directions, i.e. spatial directions and time directions. New voice data collection methods with handhelds and GPS also require such functionality.

Further suggestions for future development may be a "GRASS quality state and update system", a tool which enables the user to check the GRASS site for updates. After changing the GRASS software organization from a monolithic package to topic-oriented subsystems, it will be easier to support such selective updates. Code pieces may be locally compiled/installed as it is known from "R extensions" (R is free statistical software). Documentation should be distributed with the source code and merged on the fly into the GRASS documentation system. A backup of replaced module(s) will optionally be saved.

**A code development wish list may comprise:**

- Improved code modularity by splitting GRASS into task oriented packages, helping users to orientate themselves in a smaller system and also allowing developers to focus on code portions (since the entire code base consists of more than one million lines of code). Packages such as *GRASS-core*, *GRASS-base*, *GRASS-vector*, *GRASS-image* will allow to run GRASS with minimal system requirements and help local customization;

- There is a need to simplify the integration of existing programs (written in C, C++) with GRASS. GRASS development should concentrate on core functionality and incorporate specialized methods from tightly coupled external packages. Along with the required GRASS packages the user could run such models even stand alone;

- Libraries re-factoring and miniaturization is needed to improve code maintainability as well as to optimally support handheld devices. Internal cohesion of GRASS libraries and functions should be higher, redundancies removed (this requires clone detection). Initial work has been started [6];

- Support for distributed computing or parallelization (MOSIX cluster, Beowulf or other approaches) on a library level, enabling GRASS to run on GRID (for a possible approach see [9]);

- A GRASS daemon may manage metadata and control external connections to the system. It would be a step towards building a more integrated GIS and help create a more dynamic multi-user system;

- Better code documentation is needed, the documentation should be maintained within the code, not separately. As it will be impossible to manually maintain the *Programmer's Manual*, descriptions should remain inside the source code and be extracted on demand (e.g. into a HTML/PDF file along with general descriptions).

While some initiatives responding to this wish list have already started, it is obvious, that most will require a major effort and dedicated developers. The implementation will be to the great extent dependent on the developers needs and interest and substantial speed-up can be achieved by extensive use of experience and results from other Free software projects.

## 4 The need for Free Geo-Data

While in U.S.A. a wide range of geospatial data is freely provided through federal, state and local government agencies (for example, USGS, EPA) access to GIS data in Europe is mostly restricted. Comparison of geotechnology developments in USA and Europe demonstrates that restrictions in availability of georeferenced data slow down the development and limit the information available to citizens about their environment. Not only in terms of civil protection access to data is important. The more people know about their environment and consequences of their actions in terms of environmental impact, the awareness for nature will rise.

Scientists in Europe are often using data from U.S. systems such as GPS (the upcoming European Galileo system (planned for 2008) will partially be encrypted), monitoring the surface of Earth by satellites such as LANDSAT, TERRA (ASTER/MODIS) etc. While coastal LIDAR data are freely available in U.S., no free high resolution elevation data are provided for European countries. The only available data are the low resolution US governmental data sets such as GLOBE-DEM, GTOPO30, ETOPO5 and very limited 30m elevation data derived from Stereo-ASTER channels. While the SRTM Shuttle topography mission data were recently made available for USA, they are not yet available for Europe, Asia, South America or other parts of the world (according to JPL/DLR announcements not earlier than end of 2003, even then not freely). Also there are no European climate data accessible without paying a high amount of money while data for 11000 stations distributed in non-European parts of the world can be downloaded from U.S. websites (NCDC/NOAA). Besides raster data neither vector data are freely released to the public. It is quite surprising that even the borders of several European countries are publicly unavailable, neither from the European Commission nor from the countries' governments.

The recent floodings in Central Europe (Austria, Czech Republic, Germany, Slovakia, Romania in August 2002) which devastated many towns, affecting hundreds of thousands of people, confirm again the need of detailed knowledge about terrain structure and flow directions. These data are not made available to the public but access is restricted.

Unfortunately there is no habit or tradition to release GIS research results (even not time delayed) to the public. We believe, that the public, paying for the mapping and GIS activities, should have easy access to basic spatial data such as elevation models, climate data, and vector data networks. To protect data released to the public either by governmental institutions or private companies there is a need for a *Free GeoData License* (called "FGDL" in the sense of GNU GPL). A similar proposal was suggested for data in bioinformatics ([19]).

# 5 GRASS future

We envision that GRASS will remain a general-purpose GIS, but its improved structure, higher level of abstraction and modularity should support local adaption for specific applications. With a modularized system (in terms of code organization, internal interfaces, well defined API, etc) the GRASS modules (or sets of modules) may be implemented in different environments and stimulate further experimentation. We can expect initiatives leading to several possible paths of GRASS evolution (in Darwinian sense, with the successful ones growing and those less successful dying out):

- GRASS as GIS for everybody (Free GIS equivalent of existing proprietary systems based on free and/or commercial interfaces). This direction may be hard to accomplish with a relatively small number of developers however, GRASS may grow in this direction "naturally" as more complex applications need the efficient basic tools, too;
- GRASS as geospatial research environment which provides solid core functionality and encourages experiments and innovation;
- GRASS GIS as a professional, general purpose GIS, with reliability and stability as the most important issue;
- GRASS as a system of modules which can be plugged into specialized applications or devices to provide geospatial processing tools,
- GRASS as ??? – there is always a possibility of a completely new and unexpected direction stimulated by new technology developments and ideas

While GRASS is still catching-up for the years of stalled development a lot of effort is focusing on creating a free environment which encourages the development of new generation of tools, algorithms, data structures and applications. Open Source and Free Software infrastructure offers an excellent opportunity for efficient development of a robust and reliable GIS code with freedom for everybody to run, study, redistribute and improve it for the benefit of entire community [29].

# Acknowledgment

The support by the National Research Council and US Army Research Office is gratefully acknowledged.
Markus Neteler was partially supported by the FUR-PAT Project WEBFAQ.

## References


[1] K.J. Beven. *Distributed Modelling in Hydrology: Applications of TOPMODEL*. Wiley, Chichester, 1997.

[2] R. Bivand and M. Neteler. Open Source geocomputation: using the R data analysis language integrated with GRASS GIS and PostgreSQL data base systems. In *Proc. 5th conference on GeoComputation (CDROM), 23-25 August 2000, University of Greenwich, U.K.*, `http://reclus.nhh.no/gc00/gc009.htm`, 2000.

[3] W.M. Brown, H. Mitasova, and L. Mitas. Design, development and enhancement of dynamic multidimensional tools in a fully integrated fashion with the GRASS GIS. Technical report, US-Army CERL, 1997.

[4] K.A. Buehler and J.R. Wright. Bayesian reasoning shell for land management. I: Structure and Function. *J. of Computing in Civil Engineering*, 5(3):267–282, 1991.

[5] K.A. Buehler and J.R. Wright. Bayesian reasoning shell for land management. II: Implementation. *J. of Computing in Civil Engineering*, 5(3):283–299, 1991.

[6] M. Di Penta, M. Neteler, G. Antoniol, and E. Merlo. Knowledge based library refactoring for an Open Source project. In *IEEE Working Conference on Reverse Engineering, Richmond, Virginia, USA*, (accepted) 2002.

[7] K. Dilks. Army managing cultural resources with GRASS/Informix system. *GRASSClippings*, 7(3):33–38, 1994.

[8] K. Gardels. Defining Open GIS: An introduction to the Open Geodata Interoperability Specifications (OGIS). *GRASSClippings – The Journal of Open GIS*, 7(2):7–13, 1993.

[9] W.W. Hargrove, F.M. Hoffman, and T. Sterling. The do-it-yourself supercomputer. *Scientific American*, 265(2):72–79, 2001.

[10] J. Hofierka, J. Parajka, H. Mitasova, and L. Mitas. Multivariate interpolation of precipitation using regularized spline with tension. *Transactions in GIS*, 6(2):135–150, 2002.

[11] International Institute of Infonomics and Berlecon Research GmbH. *Free/Libre and Open Source Software: Survey and Study*. `http://www.infonomics.nl/FLOSS/report/index.htm`.

[12] SRI International. GeoVRML 2.0 (draft) specifications. Electronic document: `http://www.geovrml.org/2.0/`, 2002.

[13] P.Y. Julién and B. Saghafian. A two-dimensional watershed rainfall-runoff model – CASC2D user's manual. Technical report, Colorado State University, 1991. Civil Engineering Report CER90-91PYJ-BS-12.

[14] J. Markoff. Microsoft and Free Software at the same show? It's true. *New York Times*, (August 10), 2002. `http://www.nytimes.com/2002/08/10/technology/10LINU.html`.

[15] H. Mitasova, L. Mitas, W. Brown, D. Gerdes, I. Kosinovsky, and T. Baker. Modeling spatially and temporally distributed phenomena: new methods and tools for GRASS GIS. *Int. J. of Geographical Information Systems*, 9(4):433–446, 1995.

[16] M. Neteler and H. Mitasova. *Open Source GIS: A GRASS GIS Approach*. The Kluwer international series in Engineering and Computer Science (SECS): Volume 689. Kluwer Academic Publishers, Boston, Dordrecht, London, 2002.

[17] NOAA. *NOAA LIDAR tool*. `http://www.csc.noaa.gov/crs/tcm/index.html`.

[18] OpenGIS Consortium. The OpenGIS Geographic Markup Language (GML) Specification - november 27, 2001. Electronic document: `http://opengis.org/techno/specs.htm`, 2001.

[19] Tim O'Reilly. In the trenches: What's next for Linux and Open Source? *Linux Magazine*, October 2001. http://www.linux-mag.com/2001-10/trench_01.html.

[20] P. Ramsey. Technology trends: Open Source GIS fights the three-horned monster. *GEO World*, August 2002. http://www.geoplace.com/gw/2002/0208%5Fgw/0208gis.asp.

[21] E. Raymond. The cathedral and the bazaar. Electronic document: http://www.ccil.org/~esr/writings/cathedral-paper.html, 1997.

[22] E. Raymond. *The cathedral and the bazaar. Musings on Linux and Open Source by an accidental revolutionary.* O'Reilly & Associates, Cambridge, 1999.

[23] C.C. Rewerts and B.A. Engel. ANSWERS on GRASS: Integrating a watershed simulation with a GIS. In *ASAE Paper No. 91-2621*. American Society of Agriculatural Engineers, St. Joseph, Missouri, 1991.

[24] B.D. Ripley and K. Hornik. Date-time classes. *R News*, 1(2):8–11, 2002. http://cran.r-project.org/doc/Rnews/.

[25] D. Schell. Greetings from the Open GRASS Foundation. *GRASSClippings – The Journal of Open GIS*, 6(3):3, 1992.

[26] D. Schell. The Open GIS Foundation. *GRASSClippings – The Journal of Open GIS*, 7(2):4–5, 1993.

[27] R. Srinivasan and J.G. Arnold. Integration of a basin-scale water quality model with GIS. *Water Resources Bulletin*, 30(3):453–462, 1994.

[28] R. Srinivasan and B.A. Engel. A knowledge based approach to extract input data from GIS. In *ASAE Paper No. 91-7045*. American Society of Agriculatural Engineers, St. Joseph, Missouri, 1991.

[29] R.M. Stallman. Free Software: Freedom and Cooperation. *Transcript of a speech delivered at New York University in New York*, 2001. http://www.gnu.org/events/rms-nyu-2001-transcript.txt.

[30] FreeGIS team. The FreeGIS Project. Electronic document: http://www.freegis.org/, 2002.

[31] USGS. *National Elevation Data set (NED)*. http://edcnts14.cr.usgs.gov/Website/store/viewer.htm.

[32] USGS. *USGS Real-time Water Data*. http://water.usgs.gov/realtime.html.

[33] B.E. Vieux, V.F. Bralts, L.J. Segerlind, and R.B. Wallace. Finite element watershed modeling: One-dimensional elements. *J. of Water Resources Planning and Management*, 116(6):803–819, 1990.

[34] D.A. Wheeler. Why Open Source: Look at the numbers! Electronic document: http://www.dwheeler.com/oss_fs_why.html, 2002.

[35] H.C. Zhuang. Image processing in GRASS. *GRASSClippings – Geographic Resources Analysis Support System Newsletter*, 6(2):10–11, 1992.

## References from GRASS users conference 2002 proceedings

Proceedings editors: B. Benciolini and M. Ciolli and P. Zatelli: *Proc. of Open Source Free Software GIS – GRASS users conference*, 2002.

[36] I. Ahmed and F. Greenaway. GIS application for land planning and management in Montserrat, West Indies. 2002.

[37] R. Bivand. Development of the GRASS/R interface – GIS and statistical data analysis. 2002.

[38] R. Blazek, M. Neteler, and R. Micarelli. The new GRASS 5.1 vector architecture. 2002.

[39] M.A. Brovelli, M. Cannata, and U.M. Longoni. Managing and processing LIDAR data within GRASS. 2002.

[40] T. Cebecauer, J. Hofierka, and M. Súri. Processing digital terrain models by regularized spline with tension: tuning interpolation parameters for different input datasets. 2002.

[41] M. Ciolli, M. de Franceschi, R. Rea, D. Zardi, and P. Zatelli. Modelling of evaporation processes over tilted slopes by means of 3D GRASS raster. 2002.

[42] M. Ciolli, A. Vitti, D. Zardi, and P. Zatelli. 2D/3D GRASS modules use and development for atmospheric modeling. 2002.

[43] A. Frigeri, C. Frederico, G. Minelli, C. Pauselli, and C. Caldarella. Identifying wrinkle ridges structures from Mars MOLA and Viking mission data: using GRASS in planetary geology. 2002.

[44] A. Frigeri and G. Minelli. Low cost PDA/GPS based field logging solution for GRASS data. 2002.

[45] J.G. Fryer and G. Scarmana. Improving DEM generation by enhancing the resolution of remotely sensed digital imagery. 2002.

[46] S.G. García. A GIS GRASS-embedded decision support framework for flood forecasting. 2002.

[47] S. Hess. GRASS on the Web. 2002.

[48] J. Hofierka, H. Mitasova, and L. Mitas. GRASS and modeling landscape processes using duality between particles and fields. 2002.

[49] J. Hofierka and M. Súri. The solar radiation model for Open Source GIS: implementation and applications. 2002.

[50] P. Löwe. A spatial decision support system for radar-meteorology data in South Africa. 2002.

[51] S. Masumoto, V. Raghavan, T. Nemoto, and K. Shiono. Construction and visualization of three dimensional geologic model using GRASS GIS. 2002.

[52] S. Menegon, S. Fontanari, R. Blazek, M. Neteler, O. Dassau, S. Merler, and C. Furlanello. Wildlife management and landscape analysis in the GRASS GIS. 2002.

[53] H. Mitasova, T. Drake, R. Harmon, J. Hofierka, and J. McNinch. Spatio-temporal monitoring of evolving topography using LIDAR, Real Time Kinematic GPS and sonar data. 2002.

[54] T. Nemoto, V. Raghavan, S. Masumoto, and K. Shiono. Development of SISGeM - an online system for 3D geological modeling. 2002.

[55] S. O'Donnell. Weather Radar enhanced flash flood forecasting. 2002.

[56] A. Panatkool and S. Laoveerakul. Geographical distributed system using GRID. 2002.

[57] P. Pauditš and M. Bednárik. Using GRASS in evaluation of landslide susceptibility in Handlovská Kotlina Basin. 2002.

[58] B. Piper, C. Ratti, and H. Ishii. Illuminating clay: a tangible interface with potential GRASS applications. 2002.

[59] J. Raghunath. Potential erosion map for bagmati basin using GRASS GIS. 2002.

[60] B. Reiter. How GRASS' development reflects Free Software history and what to expect next. 2002.

[61] C. Saul, C. Bonacito, M. Costantini, and M. Spoto. Sea-bed classification and sea-bottom mapping with GRASS in the R.N.M.M. 2002.

[62] J. Stankovic. Experimental mobile wireless GRASS based GIS for handheld computers running GNU/Linux. 2002.

[63] T. Tipdecho. Automatic image registration between image and object spaces. 2002.

[64] C. Worth and B. Schott. GIS goes mobile: Using GRASS on handheld computers. 2002.

[65] P. Zatelli and A. Antonello. New GRASS modules for multiresolution analysis with wavelets. 2002.