

# THE ARCVIEW/XGOBI/XPLORE ENVIRONMENT: TECHNICAL DETAILS AND APPLICATIONS FOR SPATIAL DATA ANALYSIS

Jürgen Symanzik, George Mason University,  
Sigbert Klinke, Swetlana Schmelzer, Humboldt-Universität zu Berlin,  
Dianne Cook, Nicholas Lewin, Iowa State University.  
Jürgen Symanzik, Center for Computational Statistics 4A7,  
George Mason University, Fairfax, VA 22030, e-mail: [symanzik@galaxy.gmu.edu](mailto:symanzik@galaxy.gmu.edu)

**Key Words:** Geographic Information System (GIS), Dynamic Statistical Graphics (DSG), Statistical Computing Environment, Linking, Interprocess Communication (IPC), Remote Procedure Calls (RPC).

## Abstract:

This article describes the dynamic linking of three software packages to allow spatial data analysis within one single software environment. The work presented here is an extension into a third direction of an existing link: The statistical computing environment XploRe has been added to the bidirectional link between the Geographic Information System (GIS) ArcView 3.0<sup>TM</sup> and the dynamic statistical graphics (DSG) program XGobi. This article presents information about the technical realization of the link and provides an example how to conduct spatial data analysis in the ArcView/XGobi/XploRe environment.

## 1. Introduction

This paper reports on a dynamic link among the Geographic Information System (GIS) ArcView 3.0<sup>TM</sup>, the dynamic statistical graphics (DSG) program XGobi (Swayne, Cook & Buja 1997), and the statistical computing environment XploRe (Härdle, Klinke & Turlach 1995). This link allows us to accomplish three things in a single environment: (1) display spatial locations and concomitant geographic variables on maps, (2) visualize these concomitant variables using interactively manipulated dynamic statistical graphics, and (3) conduct numerical statistical analyses.

GISs, in particular ArcView, are widely used for the management of multivariate spatial data bases

and are ideally suited to accomplish (1). However, current GISs are little or not supportive at all to accomplish (2) and (3). But these are the strengths of XGobi and XploRe, respectively.

XGobi has been developed for data visualization in the X Window System<sup>TM</sup> and allows the exploration of multivariate data, mostly by manipulating and displaying scatterplots in arbitrary dimensions. XGobi has been described in Swayne et al. (1997) and examples of its use, such as customer panel data, or experimental measurements of laser performance and high-dimensional objects, can be found in Koschat & Swayne (1996) and Buja, Cook & Swayne (1996), respectively.

XploRe provides users with a built-in set of statistical methods, but it can be extended through a high-level programming language and by loading functions from external libraries. XploRe has a consistent user interface on each supported hardware platform since it has its own editors for data and programs, its help system can be activated from a standard WWW browser (Schmelzer, Kötter, Klinke & Härdle 1996), and it can be accessed remotely through a Java-based interface (Kötter 1997). Version 3 of XploRe has been described in Härdle et al. (1995) and an update on the current XploRe 4 can be found in Kötter (1996) and Klinke & Kötter (1996).

The link among the three programs described here extends the ideas of the bidirectional link between ArcView and XGobi (e. g., Cook, Majure, Symanzik & Cressie (1996) and Cook, Symanzik, Majure & Cressie (1997)). The addition of XploRe now gives us the following functionality: We are capable to directly issue and use XploRe commands (such as to smooth a given data set) from within another program, we can interactively exchange data among these three programs without manually manipulating external files, and we can link different types of views available uniquely in each program.

---

*ArcView 3.0* is a trademark of Environmental Systems Research Institute, Inc.

---

*X Window System* is a trademark of MIT.

The idea to link geographical and statistical packages is not new and has been discussed in several places and many software solutions have been developed. Examples that involve either ARC/INFO<sup>TM</sup> or ArcView and a statistical package are the links to STATA, SpaceStat, a software system for interactive exploration of spatial data, and the S+Gislink, described in Scott (1994), Anselin & Bao (1996), Haining, Ma & Wise (1996), and MathSoft (1996), respectively. For a more comprehensive overview see Symanzik, Majure, Cook & Megretskaia (1997). Additional examples have been presented in this special contributed session at the JSM and can be found in these proceedings.

However, to our best knowledge, this is the first attempt to link three such packages. A first description of the ArcView/XGobi/XploRe environment can be found in Symanzik, Kötter, Schmelzer, Klinke, Cook & Swayne (1997).

In Sections 2, 3, and 4 of this paper, we describe the ArcView/XGobi, the XGobi/XploRe, and the ArcView/XploRe links, respectively. In Section 5, we provide an example how these links can be used for spatial data analysis. We finish with a summary and overview on future directions in Section 6.

The software has been developed and tested on Sun<sup>TM</sup>/Sparc<sup>TM</sup> workstations, SGI<sup>TM</sup> workstations, and DEC<sup>TM</sup> alphastations. The ArcView/XGobi part of the link can be downloaded from the WWW at the URL <http://www.gis.iastate.edu/XGobi-AV2/XGobi-AV2.html>. Currently, XploRe for UNIX<sup>®</sup> systems is available for free from <http://wotan.wiwi.hu-berlin.de/xploRe/Index.html>.

## 2. ArcView/XGobi

The ArcView/XGobi link (and the others links as well) are based on Remote Procedure Calls (RPCs) as the underlying Interprocess Communication (IPC) mechanism. General information on RPCs can be found in Stevens (1990) and Corbin (1991).

There exist several publications related to the ArcView/XGobi link and its main features (multivariate data, spatial cumulative distribution functions, variogram-cloud plots, spatially lagged scatterplots, and multivariate variogram-cloud plots). At this

---

*ARC/INFO* is a trademark of Environmental Systems Research Institute, Inc.

*Sun* is a trademark of Sun Microsystems, Inc.

*Sparc* is a trademark of Sun Microsystems, Inc.

*SGI* is a trademark of Silicon Graphics, Inc.

*DEC* is a trademark of Digital Equipment Corporation.

*UNIX* is a registered trademark of UNIX Systems Laboratories.

place we only want to refer to Symanzik, Majure, Cook & Megretskaia (1997) for technical details and to Cook et al. (1996) and Cook et al. (1997) for examples that range from forest health data over precipitation data to satellite images. In this link, ArcView is always the source of the data and has to be started first before XGobi can be activated from within ArcView.

It should be noted that each of the links can be used separately. If XploRe is not available, at least the ArcView/XGobi link can be used. If ArcView is not available, the XGobi/XploRe link can still be used. And finally, the ArcView/XploRe link can be used if XGobi is not available.

## 3. XGobi/XploRe

The idea to add XploRe to the existing ArcView/XGobi link came with the decision to make use of the previously used RPC technology such that only minor modifications were required in ArcView and XGobi. As a first step, the XGobi/XploRe link was built. In the following subsections we report on the required XploRe language extension, the possibility to use XploRe smoothers from within XGobi, linked brushing between both programs, and the possibility to pass data and projections from XGobi into XploRe. In this link, either program can be the source of the data and, therefore, can be started first while the other program can be activated afterwards from within the initially started program.

### 3.1 The XploRe Language Extension

Prior to its integration into the link, XploRe did not provide the RPC mechanism for IPC. We had to add the following functions to XploRe to make it a fully operational part of the link:

**rpcstartserver** starts an RPC server using a given port number. Only one server can be active at a time. **rpcstartserver** can only be called again after **rpcstopserver** has been called.

**rpcstopserver** stops the active RPC server.

**rpcstartclient** starts an RPC client using a given port number.

**rpcstopclient** stops an RPC client using a given handle.

**rpcstarttimer** starts a timer that checks for incoming RPC requests from external clients.

**rpcstoptimer** stops a timer.

**rpcsendrequest** sends a request to a given client.

**rpmlink** links an XploRe display with an external RPC client.

**systemcall** passes a UNIX command to the underlying UNIX shell.

These functions are permanently built into XploRe and, therefore, can be used for other applications as well, e. g., an XploRe mode for emacs.

### 3.2 Smoothers

We make use of the RPC mechanism to support external (XploRe) smoothers in XGobi. XGobi itself has only four built-in smoothers: the running mean, the running median, the Nadaraya–Watson kernel estimator, and a spline estimator<sup>1</sup> (see Figure 1).

Through the link, seven additional univariate smoothers that are part of XploRe can be selected in XGobi. These smoothers are the linear regression, the symmetrized  $k$ -nearest neighbour smoother, the  $k$ -nearest neighbour smoother, the local polynomial smoother, the lowess smoother, the neural network smoother, and the isotonic smoother (see Figure 1).

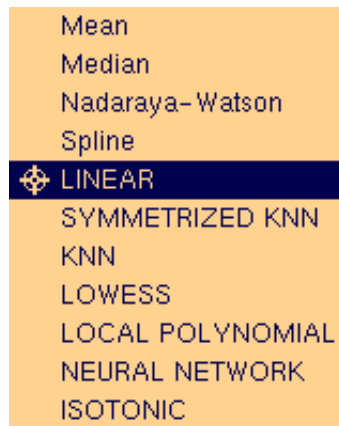


Figure 1: XGobi *Smoothers* Submenu when the link to XploRe is active. Only the four smoothers listed on top are a built-in component of XGobi, the other seven smoothers are available through XploRe.

The symmetrized  $k$ -nearest neighbour smoother and the  $k$ -nearest neighbour smoother differ in the choice of data points they use for computation. E. g., if we take  $k = 7$  then the symmetrized  $k$ -nearest neighbour smoother will take the data point itself and 3 data points left and right from the actual

<sup>1</sup>The Fortran code for the spline comes from Jim Ramsay and was taken from the package PSPLINE from the STATLIB S archive at <http://lib.stat.cmu.edu/S/pspline>.

data point. In contrast, the  $k$ -nearest neighbour smoother will take the data point itself and the six nearest data points (in the  $L_2$  sense) for smoothing.

The Nadaraya–Watson smoother is a local constant smoother (of order 0), the lowess smoother is a local linear smoother (of order 1), and the local polynomial smoother is a local cubic smoother (of order 3). For a detailed overview about local polynomial smoothers see Fan & Gijbels (1996).

The neural network<sup>2</sup> smoother is a single hidden layer network where the number of hidden neurons can be influenced. The weight decay is set to 0.001 according to Venables & Ripley (1994). In contrast to all other smoothers, this smoother may deliver different results each time it is applied to the same data. Because of a random initialization of the weights of the network we may end up after training the network with a slightly different network each time. These networks produce different predictions, especially if we only have a small number of data points.

The isotonic smoother is a monotone regression smoother (Barlow, Bartholomew, Bremner & Brunk (1972) and Hanson, Pledger & Wright (1973)) which is always monotone ascending.

Through the link with XploRe and the built-in routines XGobi now offers a broad and modern selection of univariate parametric and non-parametric scatterplot smoothers that can be applied to all static plots as well as all plots where the data is moving (e. g., during the grand tour). It is possible to smooth the data set as a whole or smooth subsets of points according to their selected brushing colors.

### 3.3 Linked Brushing

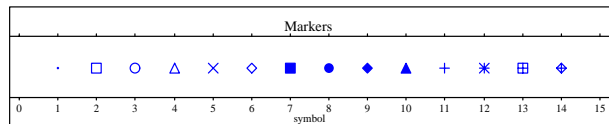


Figure 2: Plotting Symbols in XploRe. Symbol 0 represents “no point”, i. e., an invisible or deleted point, and symbol 15 represents a questionmark used to inquire points in the graphics.

The link supports linked brushing between XploRe and XGobi. We are capable to brush in an XploRe display or in the XGobi window. These views are dynamically linked. This means, if we brush a point with a new color or symbol in one view, all related views will be updated immediately. However, some restrictions apply since XploRe supports all possible

<sup>2</sup>The C code of Brian Ripley has been used.

RGB colors, 16 plotting symbols (see Figure 2), and 16 plotting sizes, while XGobi only uses 10 different colors, 6 glyphs, and 5 sizes. The following mappings have been used to match colors, symbols (in XploRe) / glyphs (in XGobi), and sizes as good as possible:

**Color:** Since XGobi just supports a set of 10 colors whereas XploRe supports a whole color palette (normally with 256 colors) in Motif we need a special routine to map colors. A color is mapped from a RGB color model to a HSV color model and a minimum weighted distance will be used to determine the “nearest” XGobi color.

**Symbols (Glyphs):** The XploRe symbols + (11), × (5), □ (2), ■ (7), ○ (3), and ● (8) are directly related to glyphs used in XGobi. The remaining XploRe symbols are mapped to Glyph 8 (a Medium ×).

**Size:** The 16 XploRe sizes (`xsize`) are mapped to the 5 XGobi sizes (`xgobisize`) according to the formula:

$$\text{xgobisize} = (\text{int}) ((\text{xsize}-2)/3.0+0.5)$$

Alternatively, XploRe sizes are calculated from XGobi sizes according to the appropriate inverse transformation.

### 3.4 Raw Data and Projections

It is possible to pass raw data and the projection currently visible (in XGobi) from XGobi into XploRe. This can be achieved by selecting the *File* menu in XGobi and the options *XploRe (pass variables)* or *XploRe (pass projection)*. Within XploRe, the data provided from XGobi is stored in the two variables *XGobiData* and *XGobiProjection*, respectively. These variables can be used similarly to any other variable created from within XploRe. However, if new data or a new projection is passed from XGobi, the corresponding variable in XploRe will be overwritten.

## 4. ArcView/XploRe

Once the ArcView/XGobi and the XGobi/XploRe links were implemented, it was relatively easy to establish the ArcView/XploRe link. The basic RPC mechanism existed within each package and could be reused without a change. Only two things had to be modified. In ArcView, the AVENUE code had to be adapted to start up XploRe (instead of XGobi) and the functions that pass data out of ArcView had to follow XploRe requirements. Also, the XploRe

code had to be modified to support ArcView’s requirements for linked brushing, i. e., an AVENUE script has to be created in XploRe and has to be executed in ArcView. What has been said about linked brushing between XGobi and XploRe holds for linked brushing between ArcView and XploRe as well. Raw data passed from ArcView is now stored in the XploRe variable *AVData*. Again, ArcView is the source of the data and has to be started first before XploRe can be activated from within ArcView.

## 5. Example

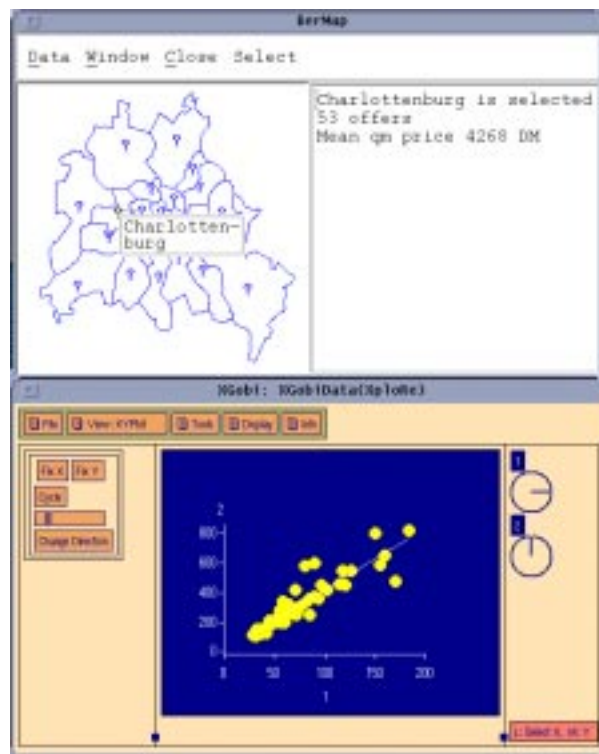


Figure 3: A link between XploRe (top) and XGobi (bottom). XploRe is used to display the geographic region of interest and a short summary statistics of a selected subset while XGobi shows a linear relationship between the area of a house ( $x$ ) and its sales price ( $y$ ), including a regression line calculated in XploRe.

In this section we look at an example where XploRe is the source of geographically referenced offers of real estate from Berlin, Germany, collected by Germer & Neudeck (1994). First, an interactive map has been created within XploRe. Then, the region of Charlottenburg has been selected (using XploRe’s “Brush” option) as the interesting region for further analysis. Within XploRe, the corresponding subset has been determined and has been sent as input to

XGobi. Within XGobi, variable number 1 ( $x$ -axis) represents the area of a house (in  $m^2$ ) and variable number 2 ( $y$ -axis) represents its sales price (in 1000 DM). Clearly, a strong linear relationship can be detected. Finally, within XGobi's *Smoothers* submenu (described in Section 3.2) we selected the LINEAR option, i. e., the linear regression. This smoother has been calculated in XploRe and the resulting line has been displayed in XGobi. The graphical output of this analysis is displayed in Figure 3.

Even though this example is only a linear regression problem, it is a good demonstration how the different components of XGobi and XploRe interact. It should be obvious how powerful this environment can be for spatial data analysis. A much more detailed analysis under consideration of geographic factors is possible if ArcView is the source of the statistical data and provides additional geographic information.

## 6. Summary and Directions

In this paper, we described the ArcView/XGobi/XploRe software environment where a Geographic Information System, a dynamic statistical graphics program, and a statistical computing environment have been linked. This is the first time that three such programs have been closely coupled in a way such that the user does not have to transfer files and is not even aware of the underlying RPC mechanism that is used to exchange data and commands among these programs.

The main application of this environment lies in the field of spatial data analysis, but especially the XGobi/XploRe link can be used for many other applications as well. An outline how this environment might be used successfully for satellite images has been given in Symanzik, Kötter, Schmelzer, Klinke, Cook & Swayne (1997). Other examples for the future use of this environment exist.

Currently, there exists one restriction in this environment, that is, only one of the links is fully functional at a time. Assume we start ArcView first and activate XGobi and XploRe from within ArcView. If we brush points in XGobi, we assume that they are brushed in ArcView (this happens) and in XploRe (this does not happen). This is similar to the situation described in Symanzik, Majure, Cook & Megretskaia (1997) where we describe a possible scenario where ArcView communicates with several XGobis, each of them displaying a different feature of the link (e. g., multivariate data, a variogram-cloud plot, and a spatially lagged scatterplot). We suggested a new type of linked brushing, called "hi-

erarchical" linked brushing. In such an environment, programs know about their "parents" and "children" and brushing information is passed up and down this hierarchy, thus guaranteeing that every program receives the appropriate brushing information through its parent or child. The idea of hierarchical linked brushing has to be extended (and implemented) towards an environment where ArcView, XploRe with multiple displays, and different types of XGobi communicate with each other.

Options how to further extend the XGobi/XploRe link have been described in Symanzik, Kötter, Schmelzer, Klinke, Cook & Swayne (1997). The ArcView/XploRe link would clearly profit if the user could directly issue XploRe commands from within an ArcView dialogue box, permitting the selection of variables and subsets of the data according to ArcView standards and results of queries, e. g., draw histograms of all measurements of water pollutants where the source of the measurement is a river or a lake, but not the open sea.

## Acknowledgements

The development of the ArcView/XGobi link was supported by an EPA grant under cooperative agreement #CR822919-01-0. A major part of the programming on the XGobi/XploRe link was done while Symanzik was visiting Humboldt-Universität zu Berlin in July 1997 and was supported by the Deutsche Forschungsgemeinschaft, Sonderforschungsbereich 373 "Quantifikation und Simulation ökonomischer Prozesse", Humboldt-Universität zu Berlin. Symanzik's work was also supported by a National Science Foundation Group Infrastructure Grant DMS-9631351. The article has not been subjected to the review of any of the previously mentioned agencies and thus does not necessarily reflect the view of any of these agencies and no official endorsement should be inferred.

## References

- Anselin, L. & Bao, S. (1996), Exploratory Spatial Data Analysis Linking SpaceStat and ArcView, Technical Report 9618, West Virginia University, Morgantown, WV.
- Barlow, R. E., Bartholomew, D. J., Bremner, J. M. & Brunk, H. D. (1972), *Statistical Inference under Order Restrictions: The Theory and Application of Isotonic Regression*, Wiley, London, New York.
- Buja, A., Cook, D. & Swayne, D. F. (1996), 'Interactive High-Dimensional Data Visualization', *Journal of Computational and Graphical Statistics* 5(1), 78-99.

- Cook, D., Majure, J. J., Symanzik, J. & Cressie, N. (1996), 'Dynamic Graphics in a GIS: Exploring and Analyzing Multivariate Spatial Data Using Linked Software', *Computational Statistics* **11**(4), 467–480.
- Cook, D., Symanzik, J., Majure, J. J. & Cressie, N. (1997), 'Dynamic Graphics in a GIS: More Examples Using Linked Software', *Computers and Geosciences: Special Issue on Exploratory Cartographic Visualization* **23**(4), 371–385. Paper, CD, and <http://www.elsevier.nl/locate/cgvis>.
- Corbin, J. R. (1991), *The Art of Distributed Applications: Programming Techniques for Remote Procedure Calls*, Springer, New York, Berlin, Heidelberg.
- Fan, J. & Gijbels, I. (1996), *Local Polynomial Modelling and its Applications*, Chapman and Hall, London, New York.
- Germer, T. & Neudeck, R. (1994), 'Betrachtungen des Angebotes für wohnungseigentum in Berlin', Diplomarbeit, Institut für Statistik und Ökonometrie, Humboldt-Universität zu Berlin, Germany.
- Haining, R., Ma, J. & Wise, S. (1996), 'Design of a Software System for Interactive Spatial Statistical Analysis Linked to a GIS', *Computational Statistics* **11**(4), 449–466.
- Hanson, D. L., Pledger, G. & Wright, F. T. (1973), 'On Consistency in Monotonic Regression', *Annals of Statistics* **1**, 401–421.
- Härdle, W., Klinke, S. & Turlach, B. A. (1995), *XploRe: An Interactive Statistical Computing Environment*, Springer, New York, Berlin, Heidelberg.
- Klinke, S. & Kötter, T. (1996), XploRe 4 — A Statistical Computing Environment, in F. Faulbaum & W. Bandilla, eds, 'SoftStat '95 Advances in Statistical Software 5', Lucius & Lucius, Stuttgart, pp. 113–122.
- Koschat, M. A. & Swayne, D. F. (1996), 'Interactive Graphical Methods in the Analysis of Customer Panel Data (with discussion)', *Journal of Business and Economic Statistics* **14**(1), 113–132.
- Kötter, T. (1996), Development of XploRe 4: The Programming Language, in A. Prat & E. Ripoll, eds, 'Compstat – Proceedings in Computational Statistics, Short Communications and Posters', pp. 201–202.
- Kötter, T. (1997), Interactive Interfaces of Statistical Software for the Internet, in W. Bandilla & F. Faulbaum, eds, 'SoftStat '97 Advances in Statistical Software 6', Lucius & Lucius, Stuttgart, pp. 153–158.
- MathSoft (1996), *S+Gislink*, MathSoft, Inc., Seattle.
- Schmelzer, S., Kötter, T., Klinke, S. & Härdle, W. (1996), A New Generation of a Statistical Computing Environment on the Net, in A. Prat, ed., 'Compstat – Proceedings in Computational Statistics', Physica-Verlag, Heidelberg, pp. 135–148.
- Scott, L. M. (1994), 'Identification of a GIS Attribute Error Using Exploratory Data Analysis', *The Professional Geographer* **46**(3), 378–386.
- Stevens, W. R. (1990), *UNIX Network Programming*, Prentice-Hall, Englewood Cliffs, NJ.
- Swayne, D. F., Cook, D. & Buja, A. (1997), 'XGobi: Interactive Dynamic Graphics in the X Window System', *Journal of Computational and Graphical Statistics*. Forthcoming.
- Symanzik, J., Kötter, T., Schmelzer, S., Klinke, S., Cook, D. & Swayne, D. (1997), 'Spatial Data Analysis in the Dynamically Linked ArcView/XGobi/XploRe Environment', *Computing Science and Statistics* **29**, Forthcoming.
- Symanzik, J., Majure, J. J., Cook, D. & Megretskaja, I. (1997), Linking ArcView 3.0 and XGobi: Insight Behind the Front End, Technical Report 97–10, Department of Statistics, Iowa State University, Ames, Iowa.
- Venables, W. N. & Ripley, B. D. (1994), *Modern Applied Statistics with S-Plus*, Springer, New York, Berlin, Heidelberg.