Integrating GIS technology with RiverWare

Functional Requirements

Author: Patrick Lynn

Graphical information systems (GISs) support storage, analysis, and display of spatial data. While most types of RiverWare objects represent physical objects that are located in space, RiverWare does not represent or make use of information about the spatial extent and location of the modelled objects. This document motivates and describes requirements for the use of GIS technology within RiverWare.

1 High Level Requirements

In this section we provide a brief description of the immediate goals for incorporating GIS technology into RiverWare; in subsequent sections we motivate and describe more detailed requirements. The new functionality will support a map based view of the modelled basin. By providing an intuitive and information-rich view of a basin, the map-based view will complement the existing schematic views.

1.0.1 Support georeferenced objects

RiverWare will support the association of *spatial coordinates* with each object. The spatial coordinates for an object will be represented as a pair of floating point numbers, and will be interpreted as either the object's geographic coordinates (latitude and longitude) or cartesian coordinates in a projected coordinate system (easting and northing, or x and y). The user will be able to provide objects with spatial coordinates in the following ways:

- By interactively entering the two numbers.
- Through graphical interaction with the workspace. For example, the user could drag an object from its original location on a map display and drop it at the map location corresponding to its spatial coordinates. Alternatively, the user could select a location on a map display and then indicate which object has that locations as its spatial coordinates.
- From a file in standard GIS format.

  [I recommend that we remove this last bulleted requirement because this represents a significant amount of software effort for little payoff. This is a one-time operation for each model, and I doubt that many of our users would take advantage of this feature.]

1.0.2 Support a geospatial workspace view

In the geospatial workspace view object icons, links, and a background image are displayed in a spatially consistent way. That is, the background image is taken to be a map projection and each object icon is displayed at (or near) the location on this map which corresponds to the object's spatial coordinates.

RiverWare will be able to read and display background maps in standard image and GIS formats. Note that some formats support information about the image's location in a coordinate system, and some do not. Appropriate mechanisms will be provided for user input (and persistence) of this information for images that do not already contain it.

[Are any of our users likely to have images in a GIS format that contains projection information? Which formats?]

1.0.3 Design for additional integration

Current development should facilitate further integration of GIS technology with RiverWare in the future.

2 Description of the current workspace functionality

In this section we discuss the purposes currently served by the RiverWare workspace, so that we can evaluate how GIS technology might improve or complement this functionality. Note that the main RiverWare dialog provides menu-based access to most RiverWare functionality in addition to providing the workspace display discussed here.

- **Schematic visualization of the modelled river basin**. This display indicates how the basin has been abstracted and discretized by the user into objects, and for each object shows the name, type, and relationships with other objects (i.e., links indicating information sharing between objects).

  In this schematic view, each type of object has a distinct, predefined icon (image) associated with it, each object in the model is represented by an icon of the appropriate type, and each icon is labelled below with the object's name. The user can manually arrange the object icons and the resulting object locations are saved within the model file.

  Links are drawn as lines between icons, and the color and style of these lines can be controlled by the user. Links can be organized into groups.

  Users control the focus of the workspace by scrolling and the level of detail by zooming in and out.

  Two types of schematic views are supported, Simulation and Accounting. The two views are conceptually similar, but the accounting view displays additional object information pertaining to water ownership (e.g., which accounts are associated with an object and what are their types).

  The object icons in each view are different, their locations in each view can be different, and RiverWare makes no assumption about the relationship between an object's location in either view and the object's actual, physical location (if any).
- **Access to more detailed information organized by object**. This information is accessed by double-clicking on the object's icon, and includes object data and methods.
- **Support for model creation**. The user interacts with the workspace to create and locate objects as well as to create links between the objects.
- **Selection of multiple objects**. For actions that operate on a collection of objects, such as creating a subbasin or saving a subset of the objects to a file, the workspace provides a graphical mechanism for defining the collection of objects on which the action will operate.

Each workspace view is drawn on a conceptual "canvas" which is characterized by the following user-settable attributes:

- Size (default width = 6450, height = 6450)
- Background color
- Text color
- Canvas font size

RiverWare 5.1 added support for a background image in the simulation view. Users can specify an image in a standard format and that image is displayed as the background of the simulation view of the workspace. The image file must be in the same directory tree as the model file and its relative path to the model file is retained in the model file. If the background image is a map of the modelled basin, then the user can manually rearrange the object icons to correspond to the actual locations of the associated objects on the map. In other words, the user can use this functionality to create a simulation workspace view in which the schematic view of the model basin is overlaid on top of, and in registry with, a map of the basin. See the document “Georeferencing Simulation Objects in RiverWare: Proposal” for details and limitations of this approach as well as possible extensions.

3 GIS technology and RiverWare

In this section, we briefly discuss GIS technology and suggest ways in which RiverWare might usefully access this functionality.

3.1 Coordinate systems and projections

This section introduces some basic GIS terminology and concepts.

Locations on the earth are most often represented as a pair of coordinates representing latitude and longitude. For some applications, a third coordinate representing something like elevation relative to mean sea level is also used. Apart from this spherical coordinate system, GIS systems typically represent locations as cartesian coordinates in a plane onto which the surface of the globe has been projected.

Several standards have been developed to provide agreed-upon coordinate systems for a given area. These standard coordinate systems define a set of map projections with together cover the target area. One common coordinate system which covers the globe is the Universal Transverse Mercator (UTM) coordinate system. This system defines a projection for the northern and southern areas of each of 60 longitudinal zones spanning the globe. For example, in the projection UTM Zone 11 North, the coordinates eastings = 397,800 m, northings = 4,922,900 m, corresponds to a location in central Oregon.

Another standard coordinate system is the State Plane coordinate system which defines a set of projections for the United States. In this system, each state is divided into one or more zones, so a location is specified by a state, zone designation, and x, y coordinates.

Note that a standard coordinate system defines a mathematical coordinate system corresponding to each projection; when these two uses of the term “coordinate system” would be confusing, we use the term to refer to the standard, and use the term “projection” or “map projection” for the component coordinate systems.

Coordinates, whether they be geographic (latitude and longitude) or correspond to a map projection, are specific to a datum. A datum is a reference surface and surveyed coordinates for a set of actual points and lines. Examples of common datums include the North American Datum of 1983 (NAD83) and the World Geodetic System of 1984 (WGS84).

From “GIS Fundamentals” (Bolstad, 2008):

> Exact or approximate mathematical formulas have been developed to convert to and from geographic (latitude and longitude) to all commonly used coordinate projections. These formulas are incorporated into “coordinate calculator” software packages, and are also integrated into most GIS software. For example, given a coordinate pair in the State Plane system, you may calculate the corresponding geographic coordinates. You may then apply a formula that converts geographic coordinates to UTM coordinates for a specific zone using another set of equations. Since the backward and forward projections from geographic to projected coordinate systems are known, we may convert among most coordinate systems by passing through a geographic system.

That is, it is relatively easy to convert back and forth between geographic coordinates and coordinates from a known projection; whereas a method for direct conversion between two different projections is not generally available.

To display a part of the curved surface of the earth on the screen necessarily requires a projection, so it will always be the case that any geospatial view will have some map projection associated with it.

On the other hand, to support our initial goals, we do not need to know anything whatsoever about the projection associated with the geospatial view, rather what we require is that the geospatial view, the object coordinates, and the map all share the same projection. Note that this allows us to support the maps for which the projection is not known, which I believe will be common.

Another relevant point is that there are many ways of describing map projections, and some systems exists in multiple variations. For example, UTM zones are sometimes designated by hemispheres and sometimes by latitude band, leading to different notations for the same zone (such as, "UTM Zone 11T" and "UTM NAD83 Zone 11 North").

In the future if we need to convert between coordinate systems, RiverWare will of course need to know the details of each projection involved (including the equation describing the projection). For example, if objects locations are provided as geographic coordinates, or if the user would like to change the coordinate system of the spatial view to replace the map with a different one from a different projection.

At the river basin scale, I think the most common conversion will be between geographic coordinates and a single projection. For example, a weather service might provide rainfall forecasts in geographic coordinates, and the geospatial view might use the appropriate UTM zone projection.

At any rate, when we choose to support conversion between coordinate systems, we can start requiring that the user specify the projection for the geospatial view, map, and objects. Presumably we would identify a set of supported common projections for RiverWare to support (we will need to be able to do the conversion math, so supporting arbitrary projections is not feasible).

My bias is to avoid representing (or requiring user input of) information that RiverWare does not use for some purpose, so I suggest that initially we not collect any information about the projection.

Note that without knowledge of the projection will not know the units of the view, although the user is free to include a legend in their image.

3.2 Display

While GIS applications usually support tabular displays of data, the spatial aspect of the data allow it to be organized visually by geospatial location. Thus a primary feature of most GIS applications is the visual rendering of spatial data, i.e., the drawing of maps.

Displays can be very flexible, allowing user control over both which data are displayed at any given moment as well how those data are displayed. The data and display are typically organized into layers, and the user might have control over the following aspects of the display:

- Whether or not each layer is visible and with what level of transparency.

- How the data associated with a layer is visualized, e.g., for raster elevation data, is a contour plot drawn or are the elevations indicated on a per cell basis using a grey scale.
- Whether or not icons are associated with data, which icons and how they are presented (e.g., size, color, and orientation to the datum's coordinates).
- Whether or not labels are present, the label font, orientation, etc.
- Whether or not a legend is drawn and its appearance.

Stand-alone GIS applications can be very sophisticated in their displays, especially in how they deal with zooming in and out. For example, at different levels of resolution, an application might automatically change font size and move or omit labels.

Note that the GIS data and maps can be displayed on a screen or saved to a file. Output formats vary greatly in the degree to which the structure of the original information is retained as well as with respect to which other applications are likely to be able to understand and display files in that format. For example, the ESRI shape file format is supported by most GIS applications and allows retention of the original structure of the data. By contrast, the JPEG image format, while supported by many applications that display images, does not retain any spatial information or even the structure to allow, for example, toggling of the layers used to generate the image.

3.2.1 The clustering problem

Phil described one solution to the clustering problem, supporting callouts.

Another option is to support more flexibility in how we display icons, specifically to support smaller, less obtrusive icons, with perhaps more control over the icon labels. This suggestion is motivated by several observations:

- I think our existing, cartoonish icons look bad when displayed on a nice map. The icons seem appropriate for our other views, but I would not want to seem them on top of a map.
- Most maps address this problem with the combination of small (typed) markers and a legend, why not RiverWare? For example, Mike Basin brochures show maps with modelled objects indicated by small dots or triangles.
- Many if not most map images will have been generated for some other purpose and so might already have labels and indicators for various map features. Allowing smaller RiverWare icons and object name location would minimize conflict between the background image(s) and our schematic.

There is no reason why we shouldn't support both solutions to this issue, though I find the callout solution complicated: for some but not all objects there are two icons, lines can be links or callout indicators, the user needs to place both the object itself and its icon.

Note that there is a third solution to the callout problem, one which the users will probably frequently use, which is simply to give clustered objects spatial locations that allow for a clear display, deviating somewhat from their actual location at times.

3.3 Analysis

Analysis of spatial data has many application to water resources modelling. For example, contour information in the form of digital elevation maps (DEMs) can be used to compute direction of stream flow, determine upstream/downstream relationships between reservoirs, compute rainflow runoff, or delineate catchments. The ability to perform, or import the results of, these and similar analyses could be useful within RiverWare.

3.4 Data access

Spatial databases employ both vector and raster representations of spatial data, and support the association of attributes (non-spatial data) with each datum. Most spatial data are stored and shared in one of a small number of formats (e.g., USGS DLG format, ESRI shape file format).

To the extent that the attributes and object abstractions represented within a spatial database match those required by a RiverWare model, that database could be used to help provide information to a RiverWare model. For example, a GIS database might contain representations of reservoirs in a basin with attributes which include the reservoir name. RiverWare could use such a database to provide objects with their spatial coordinates for the geospatial workspace view, as well as to initialize other modelled object attributes. RiverWare's namemap capability could be used to facilitate access to this information by relaxing the requirement that names in the database match those in the RiverWare model.

Current RiverWare DMI functionality allows RiverWare to access series data in spatial databases, though we could consider creating new direct database connections for specific database formats.

4 Phased development of GIS display technologies within RiverWare

In this section I present one proposal for phased integration of GIS within RiverWare. I focus on the use of GIS technology in the workspace display because I think other uses of GIS within RiverWare require relatively independent lines of development and so don't affect design decisions we are making now.

For example, to support the immediate goals and make decisions about future workspace issues, we don't need to think too much about what sort of spatial analyses we might want to support in the future. The main commonality or shared development is the georeferencing of objects, something that is at any rate definitely a part of the immediate work.

4.1 Phase 0: Simulation View Background Image (done)

In this phase, RiverWare will support the display of a single background image without knowledge of its projection. This functionality is currently supported for the simulation view, and permits the display of a schematic view overlaid in registry with a map.

4.2 Phase 1: Geospatial View (immediately)

In this phase, RiverWare meets the immediate goals, which are: 1) georeferenced objects, 2) add a view in which objects are shown at their locations on a map, and 3) design for future GIS integration. This will be accomplished by the following new functionality:

- Introduce the geospatial view which displays a schematic diagram over a map.
  - Allow the user to specify the coordinate system for the spatial view by providing the coordinates of the lower left and upper right corners (or coordinates of lower left and the size).
  - Support a background image for the spatial view

    In addition to specifying the file containing the background image, the user also locates the image in the workspace by providing coordinates for two points on the image.

    A reference to the image file persists in model file

- Provide an operation to size the workspace view canvas to the minimum size that encompasses the objects and the background image.
- Add to SimObj a pair of floating point spatial view coordinates.

  Provide an interface for the user to type in the coordinates.

  Provide a mechanism for specifying an object's spatial view coordinates by dragging and dropping its icon in that view.
- Support some degree of control over the display of icons in the geospatial view

  At a minimum, allow the users to choose between the normal icon set and a small icon set, and to choose the orientation of the label with respect to the icon. Note that eventually we might want to allow this flexibility on a per-object basis, but initially we could orient the labels in the same way for all objects (as we do now).
- Current mouse coordinates could be displayed in the task bar.

Note that an object's spatial coordinates are best thought of as the location of the object's icon, not necessarily the location of the physical entitity (if any) represented by that object. Physical objects are often larger than appropriate for such a point representation and we want to avoid confusion of this spatial location with the full spatial information contained in a database. By choosing a point representation to georeference objects, we are not limiting ourselves with respect to GIS database representations of objects, be they raster, point, line, or polygon.

We might want to consider supporting image files that contain both the image itself as well as coordinate system information in some form, allowing us to place it automatically within the geospatial view.

Open issue: do we use the Simulation View's system for embedding image references (path relative to model file)?

With the functionality developed in this phase, RiverWare will be able to display a schematic view overlaid in registry with a map. The background image (map) can be changed to another with the same projection without requiring any change to the spatial locations (geospatial view coordinates) of the objects. In this scenario, RiverWare assumes that the geospatial view, the object spatial locations, and the background image(s) are from the same projection, but RiverWare does not need to know anything about that projection. Note that geographic coordinates (latitude and longitude) would not be appropriate spatial locations for the objects, as RiverWare would not know how to translate those into projection coordinates.

4.3 Phase 2: Layers (future)

In this phase, RiverWare will support the organization of the geospatial view into an ordered set of layers.

- Users can control (at least) the visibility and transparency of each layer, and they can easily rearrange the layers.
- Allow an arbitrary number of background images, each constituting a different layer. The user will need to provide information on how each image is located in the spatial view. Since it is likely that multiple images will share not only the same projection (as required), but also location within that projection (what Phil refers to as scale and scope), we should provide good support for this case. This could be done, for example, by providing projection information from a previous layer (image) as the initial default when the user adds a new layer (image).
- One possibility: a layer for each object type, each link group.

  Enhance user control over the schematic view, i.e., for each layer, what icon should be used its objects, label orientation, etc.
- We might want to introduce a key (legend) to the geospatial display.

This work could perhaps leverage off of the current display groups, and could potentially be extended to all views. Note that all only some image formats would be appropriate for layering, those that provide representational support for transparency sucha PNG (but not JPEG or GIF).

4.4 Phase 3: Map projection sophistication

In this phase, RiverWare would add support for representation of the coordinate systems associated with the geospatial view, object spatial coordinates, images, and spatial data.

- Projection knowledge can be used to support conversion between coordinate systems. For example, the user could request that the geospatial view display object coordinates in geographic coordinates even though the map projection is from UTM Zone 7N.
- Knowledge about a projection might include wording (e.g., “easting” versus “x”) units, and orientation.

This phase would require knowledge of GIS formats and the ability to do projection math, functionality that we will probably want to achieve by linking in 3rd party software or creating a RiverWare plugin interface and plugins which use a 3rd party library.

In theory, with knowledge of projections RiverWare could allow each of the three located entities (RiverWare objects, the geospatial view, and the map image) to use different coordinate systems. However, converting one map image to another requires sophisticated computation, and it does not seem likely that users would gain much from this sort of generality. My suggestion would be to require that the geospatial view and its maps share the same projection and that the object spatial locations be coordinates in that projection or geographic coordinates.

4.5 Phase 4: Additional GIS data access

In this phase, we will add support to RiverWare for accessing spatial data and probably for representing spatial data internally.

- Support reading of common spatial database formats such as ESRI shapefiles.
- Support display of spatial data in tabular form.
- Support interactive access to spatial data via the geospatial view.

  For example, if the user were to select a location on the geospatial view, RiverWare could then translate that screen location into projection coordinates and use those coordinates to index into the spatial data (i.e., show the elevation for that location, which data would be available from a DEM).

  As another variation, if RiverWare has read vector representations of objects in a spatial database, and the name attributes of those data correspond to modeled objects, then RiverWare can translate locations in the geospatial view to RiverWare objects. That is, if the user moused over any portion of the view on which a reservoir was drawn, RiverWare could display the name of that reservoir. This could allow the users to disable the schematic view entirely if they desired without losing any of the functionality of the view (access to open object dialogs, multiple object selection).
- Support rendering spatial data sets as a layers in the geospatial view.

  This would provide the full power and flexibility of GIS visualization within RiverWare and is likely to require some sort of tight coupling with a GIS application or library. Many issues would need to be addressed, such as would RiverWare or a GIS library interact with the user to specify display options, and how would visualization specifications persists.

  One option would be for the user to make visualization choices in a separate GIS application, then save those maps in a GIS format. RiverWare could then read the spatial data along with visualization choices and pass those along to a rendering engine, which would create the image to display as a layer. In this scenario, display options would need to be made and saved in a GIS application, and then those data would be displayed as layers within RiverWare.

5 High level software architecture

To integrate GIS functionality with RiverWare, we might want to apply one or more of the following software approaches.

- In house - we write code to do what needs to be done. This is what Riverside Technology did for their StateMod application.
- Link to a GIS library - link RiverWare to a 3rd party library with a C++ API that provides the desired functionality.
- Define a GIS plugin interface and implement one or more GIS plugins - this would support extensibility and dynamic linking, allowing only those users which use the plugins to pay their associated overhead.
- Add RiverWare as a plugin or customization of an existing GIS stand-alone tool - This is what DHI did for their Mike Basin application (using the ArcGIS application).
- Communicate in real time with a GIS application. Spatial databases might be accessed via DMIs, or in the future perhaps as a web service.

6 Open source GIS software

The following are open source GIS packages that are available on at least the Windows and Linux platforms. Many other GIS packages are more restrictive, for example, Idrisi and MapWindow are supported on Windows only, and GeoVISTA Studio consists of Java bean components.

- GRASS: Open source C stand-alone GIS application and library, supported on Window/Linux. GRASS/Qt is a library for accessing GRASS functionality from Qt developed and made available as open source by Navicon.
- Quantum GIS (QGIS): Open source stand-alone GIS application and library, supported on Window/Linux/others, uses Qt4. Documentation contains an example of accessing QGIS from other applications. From the documentation: “Required dependencies of Quantum GIS include GEOS and SQLite. GDAL, GRASS GIS, PostGIS, and PostgresSQL are also recommended.”
- GDAL/OGR: Open source C++ library for reading/writing various formats (the GDAL library supports raster formats, OGR supports vector formats).

7 Notes

How should we deal with objects that do not have spatial coordinates, including all objects of type DataObj and ThermalObj? Note that DataObj’s are in practice often associated with actual physical objects (e.g., each reservoir object in a model might also have an associated DataObj.

While RiverWare should be flexible with respect to scale, there is a range of scales which are more likely to be of interest at the basin level (e.g., maps with a 1:240,000,000 scale would likely not be very useful). Does this have any implications for the current design?

Consider relevance of the Spatial Data Transfer Standard, a standard developed by the USGS used to describe earth-referenced spatial data, designed to support transfer and use spatial data on different computer platforms.

Do we need to provide a mechanism to document the datum associated with spatial data?

What support should be provided for file references embedded in model files? Currently, we support pathnames relative to the model file, and there is a software hook to allow extension to other approaches. At a minimum, we should probably allow for full pathnames and environment variable expansion.

Are there any GIS formats for saving an image together with projection information? If so, we should consider using and/or supporting that format at a relatively early stage of GIS integration. What support is provided by ArcGIS for saving metadata along with images?

The geospatial view and images could be characterized in many possible way, among the fields to consider:

- Projection and datum - E.g., UTM NAD83 Zone 12B
- Scope - E.g., coordinates for lower left and upper right corners.
- Scale - ratio of pixel to distance at standard zoom?
- Units type and preferred units - E.g., length/m, angle/decimal degrees.
- Coordinate terminology - E.g., Easting/Northing, x/y, latitude/longitude.
- Description - Uninterpreted text.

As RiverWare's incorporation of GIS technology becomes more sophisticated, RiverWare will need to explicitly represent more of this information. Phil recommends that we provide support for units and coordinate terminology in Phase 1.

If RiverWare supports a text field description or comment associated with a map projection, then the users could use this field to record projection information when they have it, even if RiverWare is not using this information.

Phil also recommends that phase 1 support ESRI shapefile input of object coordinates. This is one potential application of a plugin interface, allowing use to move the GIS format knowledge information out of RiverWare proper.