



Easy Access to Near Real-Time Data

Looking at a methodology that generates aggregated live feeds for ArcGIS clients

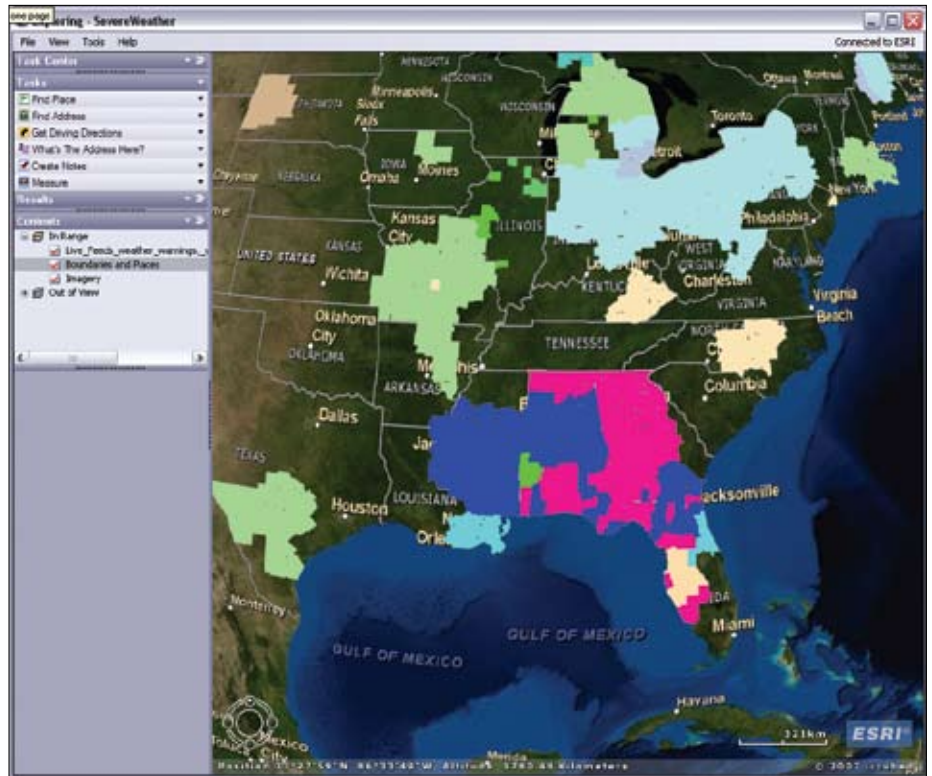
ESRI has developed a methodology that makes near real-time data and the results of analysis available through all ArcGIS clients. Called aggregated live feeds, this technique was developed by Derrick Burke and Paul Dodd of ESRI. It aggregates information captured from the Internet or Web sites; pushes that information into an ArcSDE database; and publishes this information through ArcGIS Server as a Web mapping service that can be accessed by ArcGIS Desktop, ArcGIS Server, and ArcGIS Explorer. Because these operations are performed on the server side, any client can use them without the need for custom code.

Aggregated live feeds make near real-time data and the results of analysis available through all ArcGIS clients.

Aggregated live feeds are used in applications ranging from homeland security to environmental analysis. For example, a lightweight browser application created for modeling contaminant leaks calls analysis capabilities from an ArcGIS Server geoprocessing service to perform plume modeling based on an aggregated ArcGIS Server service that contains the latest wind velocity and direction information. The analysis produces a plume that can be chained to other ArcGIS Server analyses such as identifying the demographics of the affected area.

This methodology uses simple batch scripting with a handful of public domain command line utilities to download and preprocess the feed data. The scripts use ArcSDE command line functions to push this data into the database. The scripts also incorporate logic that tracks the process to ensure they run as expected. If a load fails, an alert e-mail is sent to the administrator.

Scripts can run at varying intervals—every 5 minutes, every 30 minutes, once an hour, or once a day—depending on the application. This takes the load off clients. Continuously polling for fresh data can carry a heavy penalty, especially in browser-based applications. Processing feed data on the server side allows the client to poll for data only when needed, via standard ArcGIS Server protocols. Managing these feeds centrally can ease the demand on network resources by using a few systems



Near real-time data, such as this severe weather warning information, can be made available to ArcGIS Explorer and other ArcGIS clients, without requiring custom coding, using aggregated live feeds.

Name	Description	URL
ASC2SDE	A tool that is part of the SE_TOOLKIT by Vince Angelo. This tool will load ASCII files (such as comma-separated values [CSV] files) into ArcSDE.	arccscripts.esri.com/details.asp?dbid=14767
BLAT	A full featured command line e-mail tool used to send alerts when a load script fails. BLAT is also used in the Service Monitor.	www.blat.net/
KML2SHP	A tool written by Srinivasa Rao that converts Google KML files to shapefiles.	arccscripts.esri.com/details.asp?dbid=14980
sed	Stream Editor used to edit text files through the command line.	www.gnu.org/software/sed/
WAIT	A command line tool that will wait for another process or a specific number of seconds. It is used to pause a script and monitor the ArcSDE sdemon-o status command through the Service Monitor.	dpotter.net/Technical/2007/09/utility-to-wait-for-a-process-waitexe/
GNU Wget	A powerful HTTP, HTTPS, and FTP command line tool used to download data from the Internet.	www.gnu.org/software/wget/index.html#downloading
XML2CSV	Exports XML formatted content to a CSV file.	www.a7soft.com/xml2csv.html

Resources



Easy Access to Near Real-Time Data

Continued from page 37

Web sites or Web services. Using conditional logic capabilities, the script can be directed to restart services using logic provided by an administrator. ServiceMonitor checks Web Service Description Language (WSDL) and URLs to make sure that the site responds when it should. The new version also checks REST endpoints and ArcSDE. Service Monitor can be downloaded from ArcScripts at www.esri.com/arcscripts.

This methodology data is designed to work with previous, current, and future releases of ESRI software.

This methodology for providing near real-time data is designed to work with previous, current, and future releases of ESRI software. The approach developed by Burke and Dodd uses a variety of samples, scripts, batch files, and utilities. Many of these utilities and command line tools are available as native OS executables (either DOS or UNIX based). Aggregated live feeds developed by ESRI

Utility Aids Developers and ESRI

Generate API usage statistics on dependencies, versions, and calls

.NET developers who are developing against ESRI application programming interfaces (APIs) now have a new utility that will help them gain a better understanding of the ESRI libraries and objects.

ESRI API Evaluator, published with ArcGIS 9.3, gathers information by scanning executable (.exe) or dynamic-link library (.dll) files for specific usage of .NET types released by ESRI. It scans ArcGIS .NET code and generates a complete set of API usage statistics that will reveal the DLL dependencies; version information; and the number of calls made to all ESRI assemblies, interfaces, and members.

There are two main aspects of the utility: the Scanning wizard for defining the scan and the Results Viewer window for visualizing the results. Rules can be defined to include or exclude specific folders from scanning. Automatic upload of results can be specified in the prescan summary screen.

The Results Viewer window shows a representation of the scan results and provides a breakdown of the .dll and .exe files that are using ESRI APIs. Drilling down into the results provides information on API usage right down to a count of ESRI methods that have been called. Results are also available as a zipped XML file.

Scan results can be uploaded to a Web service hosted by ESRI for further analysis. Evaluating developer utilization of the current APIs will help ESRI better understand the needs of its users. The count of method and property calls to ESRI APIs, a list of implemented ESRI base classes and interfaces, assembly names, and contact details (if supplied) are uploaded to ESRI. ESRI will use any results from the evaluator to inform the design of future products for GIS developers.

Obtain the ESRI API Evaluator from ArcScripts at www.esri.com/arcscripts. Microsoft .NET Framework Version 2.0 Redistributable Package is required for this tool.

ArcUser

The Magazine for ESRI Software Users

300,000 ESRI Customers are within your reach. Advertise today!

Maximum Exposure. Minimum Investment.



For Rates and Media Kit, visit www.esri.com/arcuser or e-mail us at ads@esri.com.

use a native DOS batch scripting, but UNIX command line, Python, or Perl could be used.

An additional back-office process employed is on-demand caching of services. See the articles "Strategies for on-demand caching" and "Updating your map caches automatically: The key to caching dynamic data" on the ArcGIS Server Blog at the ArcGIS Server Resource Center (resources.esri.com) for more information on this process.

ArcGIS developers and users have created tools that perform similar aggregated live feed functions. To implement aggregated live feeds, one should have a good understanding of ArcSDE and command line utilities, experience with ArcGIS Server, knowledge of DOS (or whatever scripting environment will be used), and a thorough understanding of the data that will be aggregated.

The sample script snippets in Figure 1 (a shapefile loader in this case) show how the commands are used to aggregate live feeds. The accompanying table lists and describes resources used to implement this methodology.

About the Developers

Derrick Burke is the technology team lead in ESRI's Technical Marketing Department and has worked at ESRI for more than eight years. He was initially in Professional Services as a developer. Currently, he works in Technical Marketing creating prototypes using new technology and developing presentations.

Paul Dodd is the GIS systems administration team lead in the Technical Marketing Department. He has more than 25 years of experience in the computer industry working with mainframe, mini, and microcomputer systems and software. For more than 11 years at ESRI, he has worked with ArcSDE in conjunction with various Oracle and Microsoft database products and, over the years, has developed several popular data-loading and monitoring utilities for ArcSDE.