

Open Source Initiative Powers Real-Time Data Streams

NASA Technology

Let's say you are an academic researcher who needs to collect data over a long period of time, on a site hundreds of miles from your base of operation. You set up a variety of sensors in the field to continuously measure temperature, soil moisture, and capture visual imagery. All of that data needs to be stored securely so that it can later be extracted and analyzed.

But reliably acquiring and storing data is harder than it sounds. Sometimes sensors fail part way into a long-term experiment, and if they aren't communicating beyond their local network, a researcher might not know they have failed until he or she comes to retrieve the data and discovers that the experiment is lost.

Obtaining timely and reliable data from another location is also a challenge. Some researchers want to set up equipment in a remote location solely to provide an alert

when conditions are right—say, if a storm develops—so that a team can head out to the field only when it matters. Unreliable data feeds can ruin these plans.

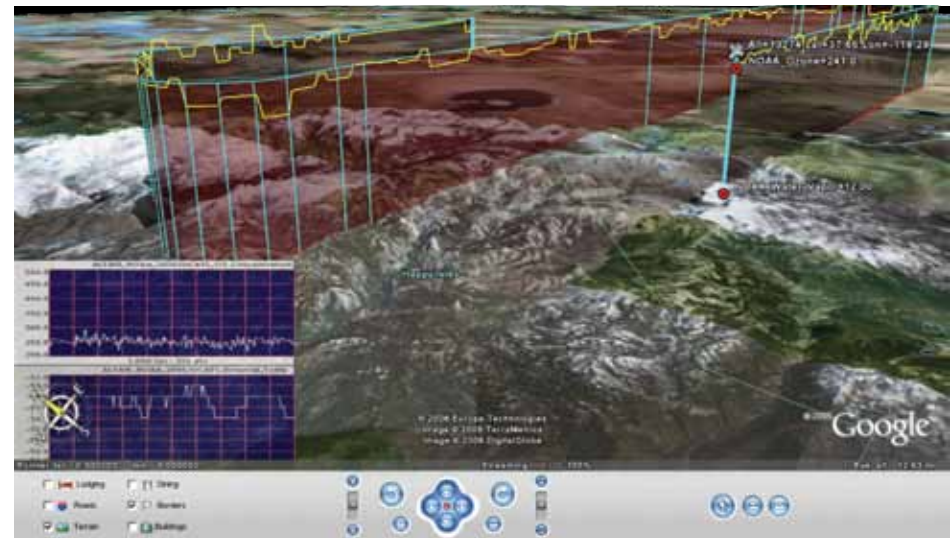
In addition, many research setups incorporate sensors from several vendors, and it is often difficult to integrate heterogeneous proprietary software and hardware into one system. Without a uniform format, it can take extra time after data collection to organize a data set with appropriate tags and identifiers.

All of these problems and more are encountered not just in academic science but in hugely diverse settings, from structural analysis to manufacturing to medical health monitoring. Tony Fountain, director of the Cyberinfrastructure Lab for Environmental Observing Systems at the California Institute of Telecommunications and Information Technology, University of California, San Diego (UCSD), has long

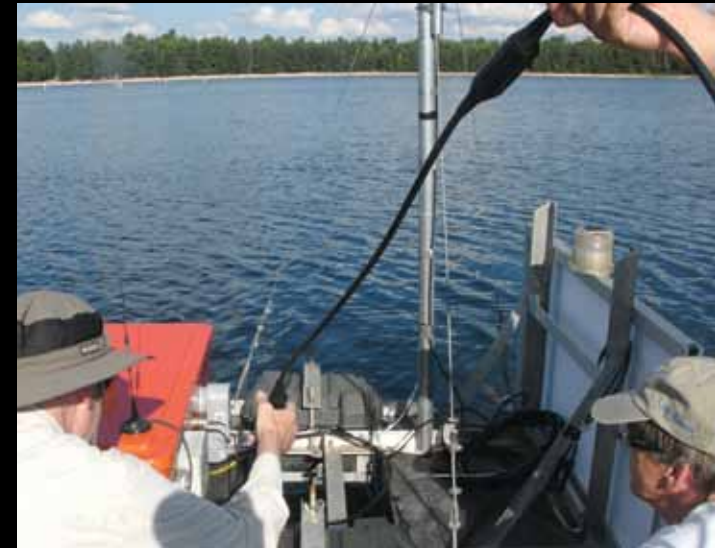
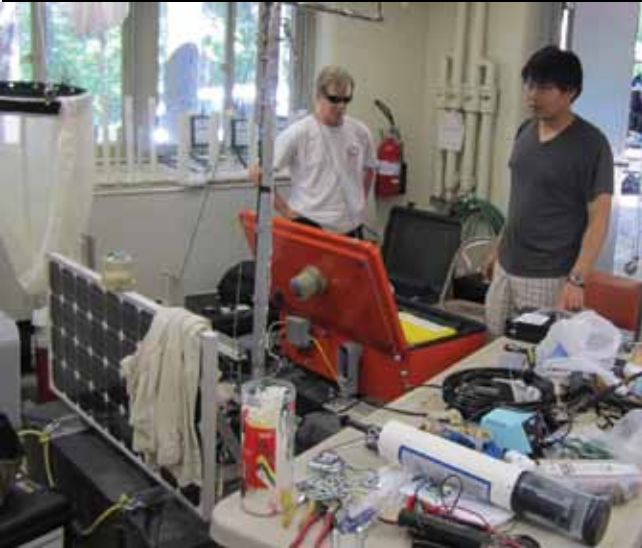
worked with researchers on securing good data. As he says, "The question is always: how do you move data between data sources, data repositories, and applications that use it? And how do you do that in near-real time, reliably, and over networks that may have various performance problems?"

At NASA's Dryden Flight Research Center, researchers have long faced the same questions. Dryden is NASA's primary center for research into every aspect of atmospheric flight and aircraft design. Many of the world's most advanced aircraft have graced the skies over the Center's location in the California desert, often with sensors attached to collect critical information.

In the mid-1990s, Matthew Miller was brought into the Dryden facility to discuss sensor data analysis software he had been developing. Miller was working as an engineer with Creare Inc. and became the principal inves-



NASA, the US Forest Service, and the National Oceanic and Atmospheric Administration teamed up to equip an unmanned aerial vehicle with sensors for detecting the severity of forest fires in California. Dryden Flight Research Center was responsible for transmitting flight data to the ground and used DataTurbine for the job.



A research team based out of San Diego releases a buoy equipped with sensors into Crystal Lake, Wisconsin. The hardware is running a version of DataTurbine ported to the Android platform, and the team has successfully streamed its real-time data across the country.

Images courtesy of Peter Shin.

tigator on a Phase I Small Business Innovation Research (SBIR) contract to apply wavelet-based mathematical analysis techniques to aircraft vibrations.

As Miller was discussing the project with the NASA technical monitor on the contract, Larry Freudinger, the two realized they had both spent time in data acquisition using devices manufactured by Hewlett-Packard (HP). “Larry happened to have some HP equipment, so we worked together to set up a demo of the software I had been working on,” says Miller. “One of its key features was that you could take data from one place and stream it live over a network to another place.” Having prepped a demo, the pair was able to get live data streaming from a test flight directly into the control room.

Technology Transfer

Both Freudinger and Miller were enthusiastic about the potential of the technology in more general applications beyond wavelet analysis and the particular equipment they had been using. Their continued efforts to develop the software resulted in a technology Miller named the Ring Buffered Network Bus (RBNB) (*Spinoff* 2000). A patent was filed by the co-inventors, and Dryden has since used the tool in many research flights.

The network streaming feature of RBNB was increasingly recognized as a significant enabling technology for distributed test and measurement applications. From its initial test run on specific machinery, RBNB soon became an open-ended tool, specializing in integrating input from any number of sources, in many kinds of formats, and moving that data to any number of end applications. A special emphasis was placed on making RBNB reliable for data storage and syncing, even when the sensor network giving it input became unreliable.

Miller, who is also a part-time contractor working for Dryden, used the partnership as a launchpad for further projects. He refined the technology through work with the Air Force, the National Institutes of Health, and the National Science Foundation (NSF), among others. All along, he had an eye on how the software could be commercialized, but he soon discovered that the tool could sometimes be a hard sell.

The challenge Miller faced was RBNB’s status as “middleware”—that is, software that is by design agnostic with respect to its inputs and outputs and which doesn’t specify how it must be used or what product it produces. Miller points out that many enterprises prefer dedicated, one-of-a-kind systems to tackle a given challenge rather


than investing in reusable, general purpose tools. “If it had been just a self-contained, off-the-shelf product, it might have been easier to sell. As middleware, it took a lot of high-tech sales effort for the payoff. Companies were reluctant to purchase it.”

Seeing “the writing on the wall,” as he says, Miller began looking into how Create might continue to benefit financially from the product other than by selling it as proprietary software. “I thought: In five years, who are you going to call for support if the software remains a privately owned commercial product that isn’t actively being sold?”

One phenomenon Miller had noticed was that the software attracted attention from groups that were interested in open source tools. For example, the Network for Earthquake Engineering Simulation (NEES), a large project created by NSF, was building structures several stories high on giant shaker tables to create earthquake simulations, and planned on using RBNB to capture its data from sensors and cameras. NEES expressed interest in being able to use RBNB as open source software.

Another proponent for going open source was Fountain, who at the time was working with a number of academic researchers at UCSD. “I’m a computer scientist,





“DataTurbine has **benefitted** the **careers** of a widely **distributed** group of people. . . . **I call that a win.**”

— Matthew Miller, Dryden Flight Research Center



The Great Barrier Reef, here shown as imaged from space, is the world's largest coral reef system and a subject of intense scientific research and study. One group in Australia is using DataTurbine to collect data gathered by sensor-laden buoys and send them to computers on shore. For a deeper look at the Open Source DataTurbine Initiative, scan this code.

but I work with ecologists, marine biologists, and others like that,” he says. “They needed environmental monitoring, and I was given the responsibility of finding the right software to do that systems integration. Our evaluation came down to two products, one of which was Creare’s.”

Miller saw that Creare might have a better commercial opportunity in providing consulting services for users of the software rather than selling the software itself. Going open source would lower RBNB’s cost, which could increase its user base—especially among the kinds of academic researchers Fountain was working with. That could in turn create demand for support services.

Miller worked with Creare to release the software under an open source license. Fountain, meanwhile, secured a grant from NSF to help its transition from proprietary software to open source and to create a website where the project could be hosted and organized. In 2007, the technology was officially rebranded as the Open Source DataTurbine Initiative—DataTurbine being an unofficial name RBNB had gone by for years.

Benefits

Today, anyone can freely use DataTurbine as a real-time streaming data engine; the software allows users to stream live data from sensors, labs, cameras, and even cell phones. The tool is as portable as the devices that carry it, and it can easily scale to match the volume of data generated by a project as it grows.

The initiative maintains a well-documented application programming interface—or the protocol required to extend the software with custom applications—so that users can adapt the program for any need. Australian researchers, for example, have adapted DataTurbine to keep track of conditions in the Great Barrier Reef, streaming live data from ocean buoys to computers on shore.

One of DataTurbine’s unique features is its ability to allow applications to pause or rewind live data streams. Users can work with data as it arrives from a lab or the

field without disrupting the process of collecting data from ongoing processes. And the data, whatever its source, comes in a uniform format. Says Fountain, “The user specifies the metadata, so once the data gets inside of DataTurbine, it all looks the same and can be relayed over continuous time periods.”

Although the software can easily be used in a variety of settings, Miller and Fountain have found that DataTurbine’s biggest user base is among researchers in the Earth and environmental sciences. Fountain works with a number of groups that have placed sensors in coral reefs, lakes, forests, and agricultural sites. DataTurbine is also playing a key role in projects that conduct large-scale simulations of seismic activity on civil infrastructure.

The software has an active development community that continues to create extensions and complementary tools. DataTurbine has been ported to the Android platform, making it tablet- and phone-friendly, and in addition many new drivers have been programmed for various classes of sensors, cameras, microphones, and

other kinds of devices. “There’s also development on the backend for data analysis and cloud computing,” adds Fountain, “so there’s a lot of development going on with mobile support.”

NASA still uses the software—for example, several years ago it equipped an unmanned aerial vehicle with it to gather information on the severity of forest fires in California. Miller continues to consult on a number of NASA projects using DataTurbine, sometimes enlisting the help of Fountain.

Both are happy to see the ways the project has grown. “Without the shift to open source, it would almost certainly be unused and unknown today,” says Miller.

“DataTurbine has benefitted the careers of a widely distributed group of people and has helped generate research grants to support scientific work. It’s been used in consulting services and has furthered other technologies,” he says. “For me personally, I call that a win.” ❖



A structure for large-scale validation of the seismic performance of bridge columns. This is one of the many projects run by the Network for Earthquake Engineering Simulation, which helped generate interest in the Ring Buffered Network Bus as an open source tool.

Image courtesy of the Network for Earthquake Engineering Simulation