Push Technology Approach to Information Streaming: ISS*Live!*

Bruce W. Hochstetler, *Tietronix Software Inc.*

ISS*Live!* is a Mission Operations Directorate project designed to deliver telemetry and timeline information in a rich content format to the public. This includes a website that hosts three-dimensional images that are fed with real-time telemetry from the Mission Control Center. In today's information technology (IT) environment, people use mobile devices such as iPads and mobile phones to access data and information content. ISS*Live!* provides this type of information from the International Space Station (ISS) to any platform suitably equipped. The ISS*Live!* project has developed a solution to the technical challenges to share ISS data with the public.

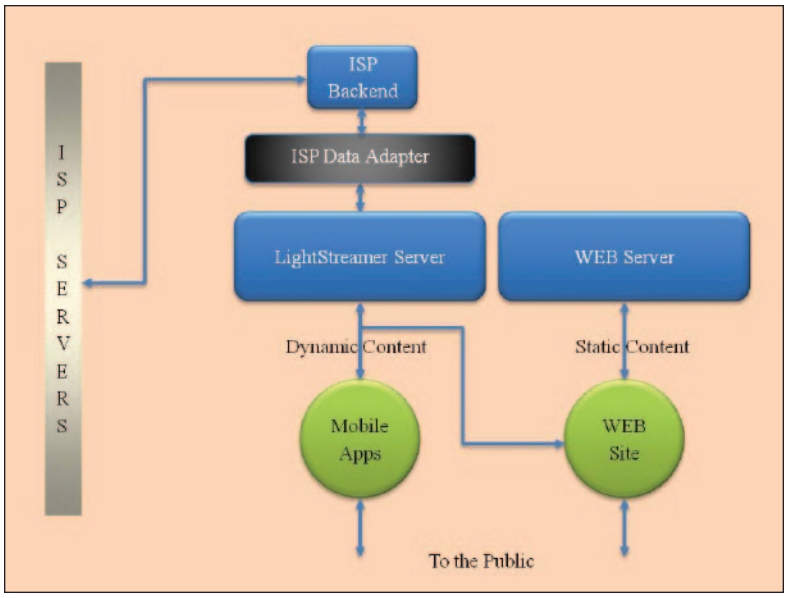

Fig. 1. *Telemetry topology.*

The classic approach fulfilling a client's content or information request uses what is known as "pull technology." With this technology, a client is required to poll a centralized server for the information on a periodic basis. The problem with this approach is that a heavy load is placed on both the server and connecting network. When multiple clients—thousands or perhaps millions of them—request information, the server must respond. In some cases, pull technology is a perfectly valid approach. For instance, when searching from a web browser, the search criteria is entered and the resultant information is returned for viewing. However, if constantly changing time-critical data are requested, this approach is not a good choice. The reason is that the client does not know when the data are new. If the polling interval is too short, the server is saturated with requests for data it does not have. If the interval is too long, the client will miss some data. It is nearly impossible to optimize the length of the poll interval.

On the other hand, "push technology" allows the server to send the data when they are ready for all interested parties. An example of push technology would be a graphical user interface that has a button. When the button is clicked, a notification is sent (i.e., pushed), indicating some action has occurred. In this scenario, a telemetry processor that delivers ISS data could also be pushed to different clients, such as web browsers, mobile devices, and desktop applications, with low overhead and significantly reduced resource requirements.

The Mission Control Center delivers telemetry using different approaches, which include both polling and pushing. The software, called the Information Sharing Protocol (ISP), uses push technology, but it does not provide a mechanism such as web browsers and mobile applications to deliver data to clients. A different solution that can leverage the ISP interface is required to accommodate these types of clients. Several commercial off-the-shelf solutions are available. Lightstreamer is one of these solutions. It was selected for its simple interface requirements, and it proved to be relatively easy to leverage the ISP interface as a data portal.

The static content of a website is delivered by the web server, while the dynamic content is sent from the push server. In this configuration, Lightstreamer can provide the push capability to different clients and avoid impacting the web server, as shown in figure 1.

Push Technology Approach to Information Streaming: ISS*Live!*

continued

While the ISP interface is efficient at processing and delivering data, it does not provide a web-based solution, nor does it provide a solution for mobile applications. ISP does have a well-defined application program interface that supports thick client development. At the time of development over 15 years ago, ISP predated the inception of mobile applications, and the world wide web was in its infancy. Consequently, ISP did not take into account these technologies. Since the delivery of the data uses the hypertext transfer protocol (http), the push server can be set up to use the secure http with the Secure Sockets Layer/Transport Layer Security. These are the same cryptographic protocols that provide communications security over the internet.

While the Lightstreamer server is primarily used for delivery of telemetry, it can be used to solve other problems that require low bandwidth with simple asyronchronous notification. For instance, most websites are backended by databases. In some cases, based on the business model, the web application may be waiting for a change to occur in the database before updating the view. The typical solution of course would be to poll the server for any changes. Consequently, this places undue loads on the database. A simpler solution would be to provide a "SqlDataAdapter" (Structured Query Language [SQL]) object that places a "SqlDependency object" on the database and the application—in this case, the data adapter. When a change is detected, the notification is sent to all interested parties. Other solutions might include the delivery of special data files (e.g., Extensible Markup Language) that an application may need to consume and process.

As NASA moves to the 21st century, the new emerging technologies are presenting new challenges and opportunities for more cost-effective system architectures. It is seen, in this example, that the push technology provides a flexible-yet-robust solution with less-intensive resource loading, hence cost savings. This architecture moves Johnson Space Center, the Mission Control Center, and NASA toward solutions that will allow the flight control team to work from anywhere, at anytime, without the geographic constraints of being physically located in the Mission Control Center. Moreover, because of the unique ability to isolate the data as one-way, this solution allows NASA, for the first time, to deliver real-time ISS telemetry to the real stakeholders and owners of the ISS—the American public.