

Data-translation server

Transcoding Publisher reformats data

■ IBM tool transfigures applications for wireless platforms, eliminating the expense and hassle of manual reauthoring

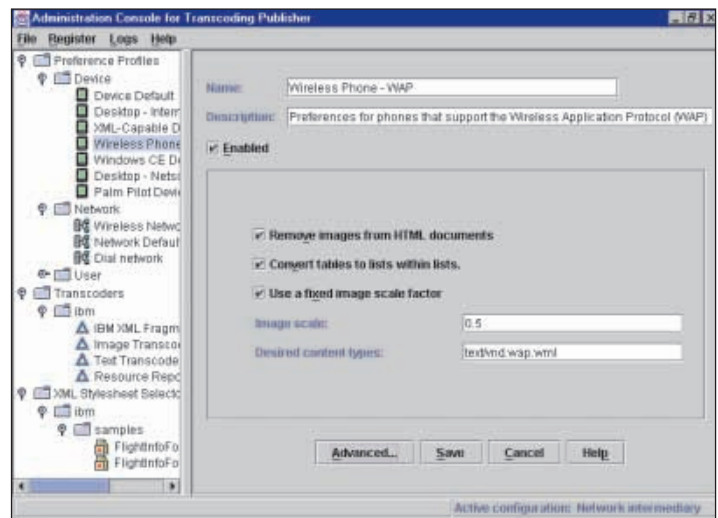
By Tim Fielden, InfoWorld Test Center

ENTERPRISE COMPUTING

COMPETING IN TODAY'S business environment means being flexible. The current move toward virtual and mobile workforces has forced companies not only to address interoperability issues between distributed and often distinct platforms but also to let users access data and applications, regardless of their physical location or choice of device. Moreover, because not all end-users are technically adept, accessing this information must be transparent and easy.

Because mobile devices tend to have smaller, low-resolution displays and limited processing power and memory, many companies and users have been reluctant to deploy them as anything more than glorified PIMs (personal information managers).

But with the recent explosion of wireless technology, companies are beginning to treat mobile devices as the crucial business communication tools that they are. Organizations now face the challenge of providing wireless platforms with the highest-quality business-critical information possible (see our Test Center Analysis of transcoding, page 57, May 29, 2000).



TRANSCODING PUBLISHER'S graphical Administrative Console allows you to easily modify preferences to define the end-user's device restrictions.

THE BOTTOM LINE

IBM WebSphere Transcoding Publisher 1.1



BUSINESS CASE

Transcoding Publisher extends existing applications and Web pages to diverse platforms, better connecting you to your mobile workers, providing easier access for wireless purchasers, and saving you bundles in development costs.

TECHNOLOGY CASE

Transcoding Publisher is a server-side translation tool that uses constraints and preferences to modify applications and Web content. It can be deployed as a network proxy, as a servlet running in the WebSphere Application Server, or as JavaBeans.

PROS

- Excellent cost-saving opportunities
- Impressive interoperability with load balancing, caching, and other tools
- Scalable
- Easy to use
- Great documentation

CONS

- Lacks support for WebSphere Application Server 3.0
- Large resource footprint

COST

\$20,000 per processor

PLATFORMS

Windows NT 4.0, Windows 2000 Server, Solaris 7 and later, Red Hat Linux 6.1, AIX 4.3.2 and later

IBM Corp.

Armonk, N.Y.
(800) 426-3333 [www.ibm.com/
software/websphere/transcoding](http://www.ibm.com/software/websphere/transcoding)

For scoring and testing methodology, go to www.infoworld.com/method.

Reauthoring existing data

Some companies resolve these access problems by reauthoring their data and applications for multiple platforms and devices. But this can be an extensive, inefficient, and costly process, especially if your content changes frequently. The reauthoring process requires modifying every application several times to fit each available wireless platform. This is a laborious and expensive task that will only become more burdensome as new devices continue to emerge.

When you keep the same data in various formats, you also run the risk of inadvertently omitting updates in reauthored versions or introducing errors during the recoding process.

Translating existing content

A better solution would be to host an application that could automatically transform existing content into different output formats. By doing this, you can expand the availability of your goods, services, information, and applications to a wider user group — and you can do so with zero recoding.

IBM WebSphere Transcoding Publisher 1.1 promises to do just that. By using this server-side legacy-to-Internet software, companies can automatically convert content from one format to another in modified versions that account for device and network constraints.

Without altering your source data, Transcoding Publisher transforms data from one single-source repository to a wide range of platforms, customizing your content for specific devices.

TEST CENTER

Many tools transform data from one format to another, such as Advanced BusinessLink's Strategi, which takes AS/400-based information and converts it to WML (Wireless Markup Language), but IBM's solution supports an unprecedented number of input and output formats.

Transcoding Publisher can handle any type of input Web data, including HTML pages from a Web server, HTML output produced from a host application such as IBM's Host Publisher, and XML data from a back-end transaction system.

Transcoding Publisher's wide selection of supported output platforms, including wireless telephones, Palms, and Windows CE devices, combined with its inclusion of expected browser-specific formats, makes it a cutting-edge solution that earns an Excellent score.

Hassle-free configuration

I installed the product without incident. My test machine had just been formatted, so I was happy to find that IBM had even included the required Java Development Kit, Version 1.1.8, and had installed it as part of the offering.

The Administration Console is the primary tool for configuring clients, and I used it to perform all configuration functions, such as creating, changing, and deleting preferences, and to handle all normal management functions, such as starting and stopping.

By adding or changing preferences, I could modify the underlying characteristics of my devices, networks, and users with no problem. Developers will be happy to hear, however, that because IBM bundles most of the popular formats with the Transcoding Publisher tool, they may never again need to add a preference or create a new format.

I was also happy to see that I could use the console to trace and even log requests for both auditing and problem determination.

Control comes at a price

For my test installation, I implemented Transcoding Publisher as a proxy and needed only to configure my clients with the address of the Transcoding Publisher server as my HTTP proxy to begin.

Transcoding Publisher transforms data prior to sending it, which not only cut down on my connection time for downloads but also gave me more control over my data transformation, allowing me to specify the amount of data I wanted transferred.

However, this server-side transformation does not come cheap: It requires a minimum of 256MB of RAM. Unless you have hefty servers, this could be an impractical drain on your resources.

How does it do it?

Transcoding Publisher transforms single pieces

of content for the requesting device, based on what it knows the requesting device can handle and on the capacity of the network being used.

When a user sends a request to the server, the server determines the requesting device-type from the user-agent value in the HTTP header and locates the matching device preference profile. Should the request come from a network, the server locates its port of entry and selects the appropriate network preference profile.

The server then modifies the request with device restrictions. For example, you might indicate in the preference profile of a particular device that it can't support JPEG images and request that all such images be recoded to GIFs.

The requested document is then retrieved in whole form from either the Web server or somewhere within the firewall. The server then executes the appropriate transformations and finally returns the transcoded document back to the requesting user.

It is also possible to tailor images to adjust screen size, file size, and number of colors, allowing users to sacrifice aesthetics for a faster download.

For instance, if I resized an image from 1,024 pixels by 768 pixels to 480 pixels by 240 pixels, it might not look as pretty, but it would be functional and would download much more quickly.

Features and deployment

The solution consists of two main parts: the Administration Console and the server. The Administration Console enables administration of the publisher, whereas the server receives requests, transcodes information, and returns data to users in an appropriate format.

I was thrilled to find that the solution includes a developer's toolkit, complete with a developer's guide, a few sample profiles, documentation for all the transcoding APIs, and tools to view transcoding requests and to validate output. This kit provides all that your developers will need to modify Transcoding Publisher itself.

The Administration Console offers a number of choices for transformations. Not only could I use the tool to manage HTML for the various flavors of browsers, but I could also convert XML documents to different XML dialects, as defined by the output DTD (Document Type Definition). Additionally, I could convert my data into WML for display on devices using WAP (Wireless Access Protocol).

You can deploy Transcoding Publisher in a few ways, either as a network proxy, as a filter (servlet) running in the IBM WebSphere Application Server, or as JavaBeans.

Should you choose to run the solution as a network proxy, you'll have the option of using it

in addition to a cache server, netting reduced server load. This provides a single service to transform information that originates from separate Web servers and also enables your users to simply point device browsers to the Transcoding Publisher server.

The downside to this approach is that the tool cannot transform data that has already been encrypted, which disallows it from transferring any secure documents back and forth between platforms.

If you require neither data transformation from multiple Web servers nor encrypted sessions, you might want to deploy Transcoding Publisher as a WebSphere filter. This deployment method allows you to choose whether or not your content is transformed before it is encrypted.

Although I was disappointed to find that the tool currently supports only WebSphere Application Server, Version 2, IBM says support for Version 3 is forthcoming.

And finally, should your shop be engaged in custom development, IBM also offers the transcoders — the actual content modifiers — as JavaBeans, enabling inclusion into existing servlets, independent content-providing solutions, and JavaServer Pages.

Fast and readable results

I tested the solution by converting an existing HTML document into WML for display on a digital telephone. Wireless service was unavailable at my location, but I verified the transformations using a 6150-telephone simulator and toolkit, both provided by Nokia.

Without making a single change to my existing HTML document, I instructed Transcoding Publisher to display the information from that page on the Nokia 6150 and was happy to receive accurate, quality, readable, and fast results.

I also retrieved stock symbols from IBM's Web site and displayed them in a similar fashion. The output was superb and I was generally very impressed with the tool's functionality.

It will soon be impossible for your company to stand up to its competition without proactively accommodating the increasing number of global wireless users. As the wireless economy unveils itself, IBM Transcoding Publisher arms you with the technology you'll need to survive. Despite its few flaws, this progressive product earns an Excellent score for its smooth performance and strategic use of sophisticated technology.

Tim Fielden is a senior analyst for the InfoWorld Test Center. Contact him at tim_fielden@infoworld.com.