

Ericsson's WebOnAir Information Server

Andy Johnston, David Kemp and Thomas Papanikolaou

The full potential of the mobile Internet is increasingly being realized. Applications and content will center on the mobility, location and situation of the user; they will become "situation-oriented." This intelligence can be used to create highly valuable, personalized services.

Ericsson's WebOnAir Information Server opens up the world of accurate, convenient and timely information to operators, service providers and users, facilitating the integration of SMS, WAP, positioning, and personalization into value-added mobile services. The server provides a robust, flexible platform and a scalable architecture for developing innovative applications that harness the power of mobility.

The authors describe the WebOnAir Information Server, highlighting its main features—the Channel Studio and development kits—which are powerful tools for the creation of complex mobility applications.

Introduction

The mobile Internet is about to enter our daily lives in a big way. It will change the way we keep in touch with our friends and family, do business, conduct our personal finances, and access entertainment. Mobile devices will be our personalized windows on the world, multiplying the range of information, services and applications available to us while we are on the move, and enhancing the quality of our lives. Ericsson is already working across several industry sectors to develop mobile Internet applications and services for mobile operators around the world.

User demands

Mobile users place special demands on information services. They demand control over how and where they receive information, and they want to be able to specify what information they need. The key to exploiting mobility as a value-added service is to have an intelligent server that always knows where its users are, what they like, and what they can be offered. Ericsson's WebOnAir Information Server (WOAIS) endows developers with the means of meeting this challenge of "situation-oriented" applications.

Meeting the challenge

The WebOnAir Information Server provides elaborate strategies for maintaining information and delivering it according to subscribers' preferences.

Mobility applications that employ these strategies place new demands on service platforms. The number of information updates and the number of requests from subscribers have the potential of multiplying

into hundreds per second if a service becomes popular. The features of the WebOnAir Information Server allow services to be dimensioned for this kind of growth.

Universal deployment

The WebOnAir Information Server allows developers to write applications that support all digital mobile phones on the market. Besides standard hypertext markup language (HTML), it offers time-based, event-based, and on-demand services over short message service (SMS) and supports the next generation of wireless application protocol (WAP) phones. Four features will pave the way for deployment to the entire user base.

- Mobile-originated short message service (MO-SMS) can be used to access information requested on any type of telephone.
- Push services will be implemented on SMS until WAP push (WAP 1.2) becomes available.
- WAP wireless markup language (WAP WML) is used to serve WAP phones, such as the Ericsson R320 and R380 and personal digital assistant (PDA) devices, such as the MC 218.
- HTML will be provided to support PDA devices that have no built-in support for WAP.

Scalability

The WebOnAir Information Server is built around TIBCO's high-performance TIB/Rendezvous messaging bus. This permits applications and system components to be distributed across nodes and networks. Such capability combined with appropriate system design allows the system to be scaled from a single inexpensive node configuration to a high-performance distributed system.

Flexibility

The architectural split between service and content provisioning allows the easy addition of content feeds (for example, including a new content provider) without affecting the functionality of a service application. This architecture enables operators to react quickly to changing market conditions, and allows developers maximum flexibility in developing long-lasting applications.

Developer productivity

More than ever before, today's competitive environment stresses the importance of short time to market. With its Channel Studio application and development kits, the Web-

BOX A, ABBREVIATIONS

ADK	Application development kit
AvisIM	Across wireless SMS transport gateway
CSS	Charging subsystem
FH	Feed handler
HTML	Hypertext markup language
HTTP	Hypertext transfer protocol
ICSS	Internal communication subsystem
ISS	Internet subsystem
LAN	Local area network
LDAP	Lightweight directory access protocol
MO SMS	Mobile-originated SMS
MT SMS	Mobile-terminated SMS
OMSS	Operation & maintenance subsystem
PDA	Personal digital assistant
PHP	Hypertext preprocessor
SA	Service application
SDK	Software development kit
SMS	Short message service
SNMP	Simple network management protocol
TCP/IP	Transmission control protocol/Internet protocol
TIB/RV	TIB/Rendezvous messaging bus
TLSA	Top-level service application
TPSS	Transport subsystem
UCP	User control point
XML	Extensible markup language
WAN	Wide area network
WAP	Wireless application protocol
WML	Wireless markup language
WOAIS	WebOnAir Information Server

OnAir Information Server enables the rapid creation and deployment of services in both a programmatic and non-programmatic fashion.

Server features

Overview

The WebOnAir Information Server provides the means of implementing a number of functions (Figure 1):

- aggregation of online information from different content providers with different formats into a pool of normalized data (extensible markup language, XML, format);
- collection of end-users' personal preferences;
- analysis of a user's current situation;
- distillation of important information at a given point in time or on end-user request;
- delivery of information to users in the desired format (SMS, WML, and HTML); and
- handover to route planning and maps if the user wants these services.

In future releases, the WebOnAir Information Server will be integrated with

- mobile positioning;
- mobile e-payment; and
- WebOnAir interactive media.

This integration will enable additional services, such as route planning, electronic payment, and multimedia transmission.

Channel Studio features

Overview

Ericsson's Channel Studio is the main application in the WebOnAir Information Server. It allows mobile telephone network operators to offer direct broadcasts of information services to millions of subscribers or to small, targeted groups of mobile subscribers (usually fewer than 50,000). Such broadcasts are not economically viable for traditional broadcast media.

Operators take information from a range of providers and transform it into channels.

Users subscribe to these channels to receive the information on their SMS- or WAP-enabled phones—at their own convenience (Figure 2). The WebOnAir Information Server delivers the information to the mobile network via an SMS center (SMS-C) or WAP gateway. No software development is required to create and deploy this service.

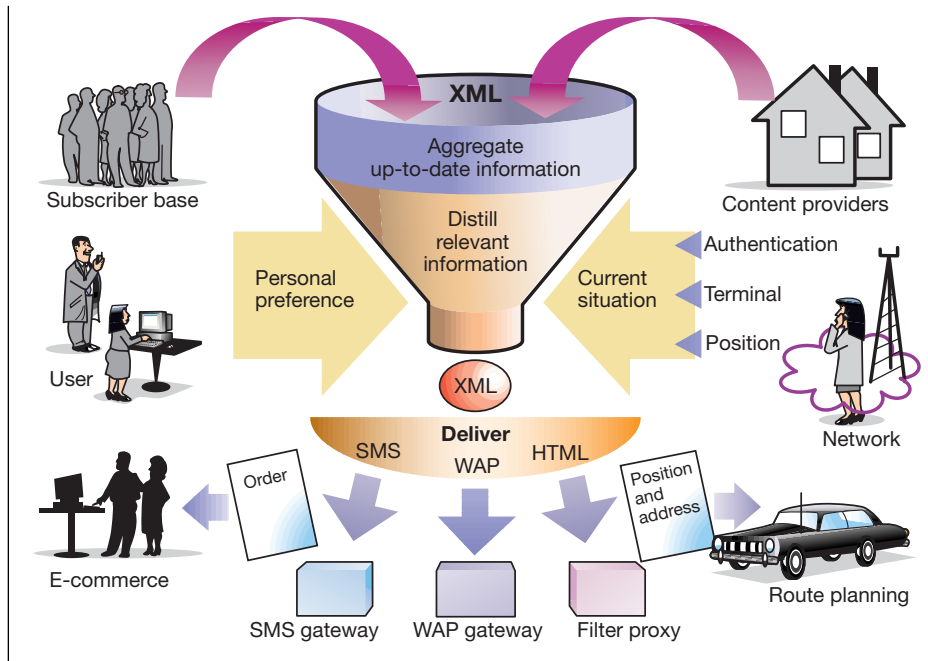


Figure 1
The WebOnAir Information Server—function chart.

Figure 2
Online information from content providers is aggregated, personalized, and delivered to the user.



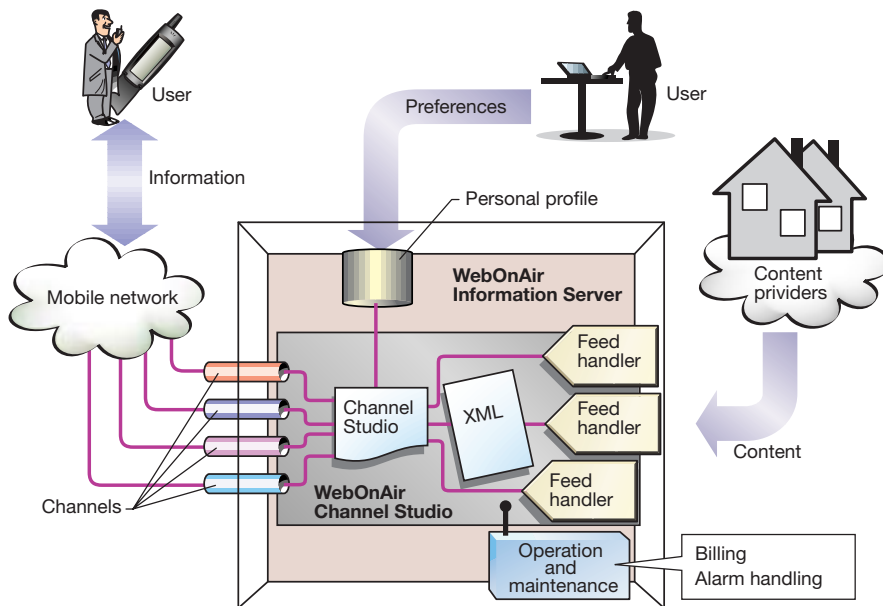


Figure 3
Channels are created when content is fed to the Channel Studio by any number of content providers.

Channel concepts

A channel is essentially information that has been grouped according to a theme, topic, or information type. The channel is created when content is “fed” to the Channel Studio by any number of content providers (Figure 3). It can include news, weather, sports information services, or specialized services that focus on specific interests, such as tour agencies, wineries, or entertainment. For example, one channel might be the “Winter

Olympics” channel, and another the “Adventure Tours” channel.

The powerful feed-handler capability of the Channel Studio is one of its major advantages for service providers because it allows channel content to be added and manipulated easily. This ease of development is essential for service providers; it enables them to maintain up-to-the-minute information on existing channels and to create new channels at a moment’s notice.

Content providers interface to the Channel Studio through the WebOnAir Information Server using a standard HTTP interface to a Web server.

A channel’s content consists of content items. A content item is an XML string that consists of either

- a short text message, called a summary, which is sent to a subscriber’s mobile phone as an SMS message; or
- an extended version (full story), which can be viewed on a WAP-enabled phone (Figure 4).

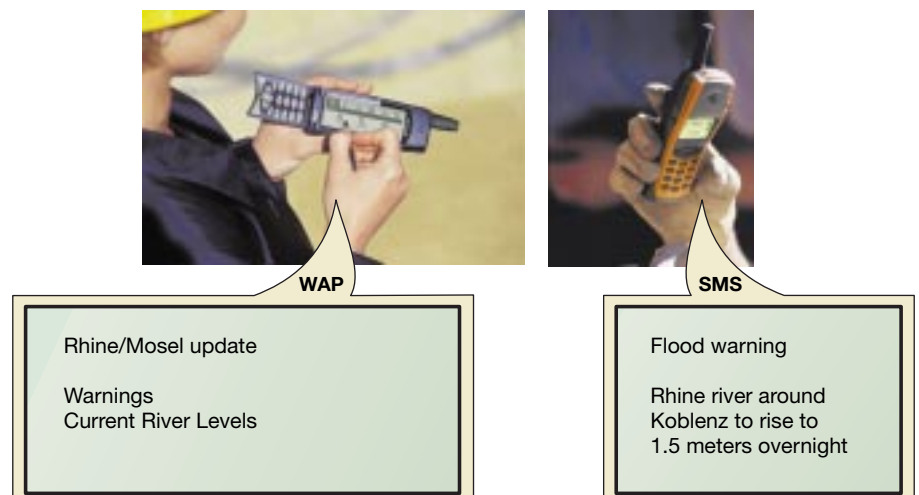
Subscribers with SMS phones can view a series of summary messages. Subscribers with a WAP-enabled mobile phone can browse through a number of summary messages and then select one of the summaries to view a full story.

Creating services with Channel Studio

Creating channel content

No programming expertise is required to create the content for a channel. The Channel Studio normally receives content via a data feed from a content provider. The con-

Figure 4
Information, displayed as a summary on an SMS display, and on a WAP display, as a series of headlines that can be selected to view details.



Content provider uses a set of simple, predefined XML tags and submits the XML content through an HTTP post operation via the Channel Studio feed handler. This method allows large volumes of information to be provided on a continual basis.

Alternatively, for smaller volumes of information, the operator or third parties can create content manually by using the Channel Studio Web interface shown in Figure 5.

Creating a channel

The creation of channels is a simple configuration process that can be performed by the operator's own staff.

Each content item delivered by the Channel Studio has a set of tags associated with it that categorize the content item in simple terms. These tags, used internally by the Channel Studio, are invisible to subscribers who use the service. For example, a news flash about the manned mission to Mars landing on the surface of the red planet might be tagged:

```
<NEWS>
<SPACE TRAVEL>
<NASA>
<MARS>
```

An operator creates channels simply by deciding which content tags should be grouped together to provide an information source. The Channel Studio organizes content items with tags into channels and makes them available to subscribers.

Structure of the content

To understand the presentation of Channel Studio content, it is important to understand the way information is organized in the world of the WebOnAir Information Server.

Information (including Channel Studio content) is organized hierarchically into two different association levels: packages and services. The Channel Studio adds two more association levels: channels and content items.

This hierarchy allows other WebOnAir services to be offered along with the Channel Studio's services in a single, cohesive package. Content is grouped according to a theme, topic, or information type. For example, a variety of different sports services can be grouped into a single package called Sports. The structure of the content is shown in Figure 6.

- **Package**—a package consists of one or more general topics; for example, sports, entertainment and news. Each topic is a



Figure 5
Content can be created manually with the Channel Studio Web interface.

service. A package is the level to which the mobile phone user subscribes.

- **Service**—a service is either another WebOnAir Information Server application or a collection of Channel Studio channels based on some association. If Sports is the package, then the services under Sports might be soccer, football, horse racing, boxing, and so on.
- **Channel**—a channel is a collection of content items based on some association. If Sports is the package and soccer is the service, the operator could provide separate channels for each of the teams in, say, a national competition. A channel can only be associated with one service, but it is possible to have multiple channels sharing the same content items. For example, content about the results of Olympic events could be fed into both a News channel and a dedicated Olympics channel.

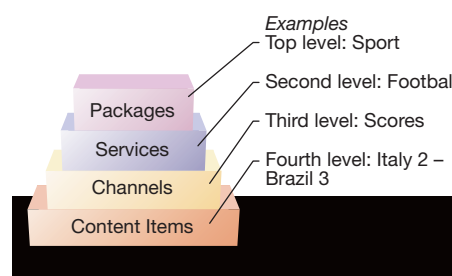


Figure 6
Information in the WebOnAir Information Server, organized into four association levels.

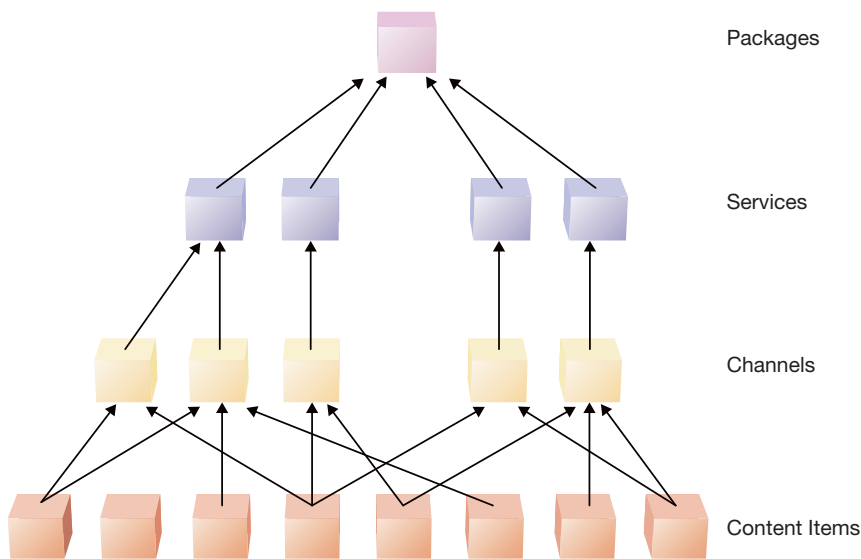


Figure 7 Relationships between packages, services, channels and content items.

- Content item—as stated above, a channel contains messages called content items. One content item can be used in several channels. For example, a content item about the weather forecast for Salt Lake City might be associated with the Salt Lake City Weather channel. But the same content item could also be used in the

Olympics channel to inform Olympic spectators about the weather forecast for the current day of the games. The method of preparing content is too detailed for this article; however, complete details are provided in the Channel Studio Operator's Guide.

Figure 7 illustrates the organization of the content and shows how the content levels are related to one another.

Using the Channel Studio

The Channel Studio gives the subscriber a range of convenient options for receiving information services and messages on a mobile phone, as well as options for delivering them to the phone.

Subscribing to channels

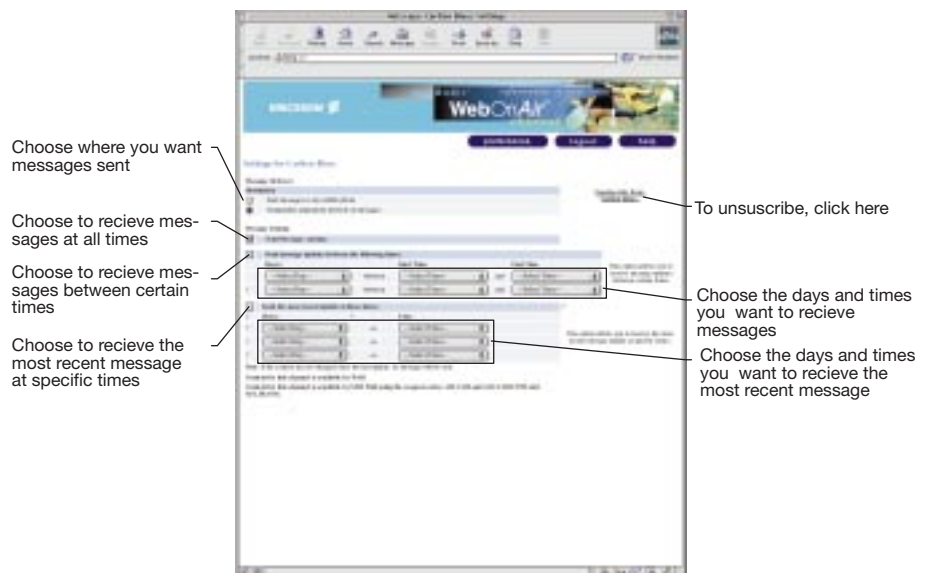
Users can subscribe to various channels using the “Channels” Web page. Users of WAP-enabled mobile phones also have the option of subscribing to a channel by selecting the “Subscribe” button on the phone display.

Choosing how information is delivered

Subscribers are given a variety of information delivery options, such as specific days, specific time intervals, and so on. These and other options are selected from the “Settings” Web page shown in Figure 8.

For the operator, being able to deliver in-

Figure 8 A “Settings” Web page for Financial News.



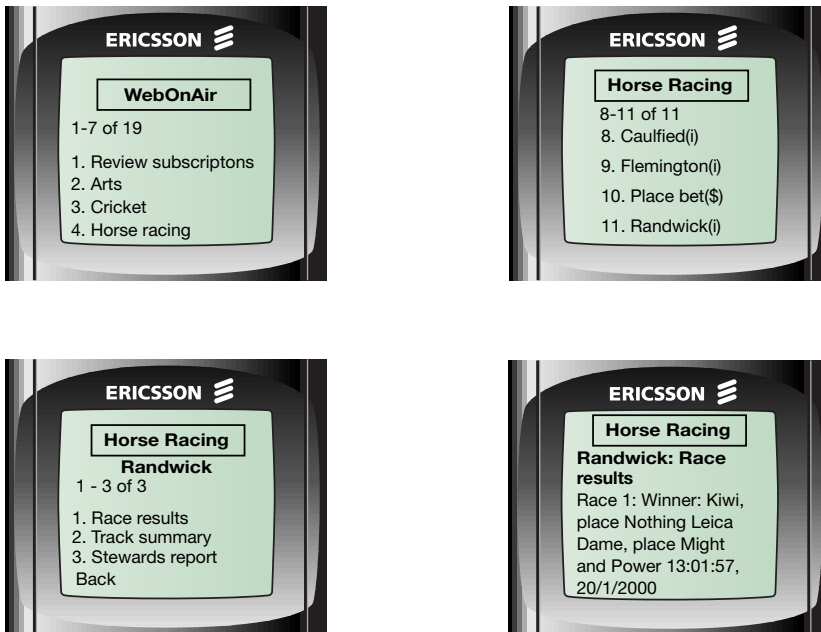


Figure 9
Package—the main screen shows the available packages, such as news, weather, or entertainment. In this example, the selected package is *Horse racing* (top, left).

Service—the next screen shows the different Races available. In this example, the subscriber selects *Randwick* (top, right).

Channel—the next screen links to the race results, track summary, and steward's report. The subscriber scrolls to and selects *Race results* (bottom, left).

Subscribing—If the user wants to subscribe, he selects *Subscribe* (not shown).

Full story—the *Race results* message is displayed on the subscriber's WAP phone (bottom, right).

formation in real time as it arrives at the Channel Studio is useful if subscribers want to keep up to date or be able to make critical decisions immediately. A financial news release saying that a certain company has just posted a record profit for the year might be a crucial piece of business information on which the subscriber wants to take immediate action.

Receipt of regularly scheduled information is also valuable when the degree of urgency is want-to-know rather than need-to-know. The Channel Studio allows subscribers to have information delivered at defined times during the day, without having to make an explicit request. For example, a subscriber may want to receive weather updates, new movie information, or similar non-critical information.

Users of SMS phones can view summary information that has been sent to their phones as short messages, either by using a request code or when the Channel Studio alerts them with new information. Subscribers can also request information at any time by sending a short request code, such as WEATHER, to a service number. The in-

formation is then returned to them in an SMS message.

Subscribers with WAP phones can browse information services by scrolling through the menus and then selecting an option. Figure 9 illustrates a WAP display for the menu selection process.

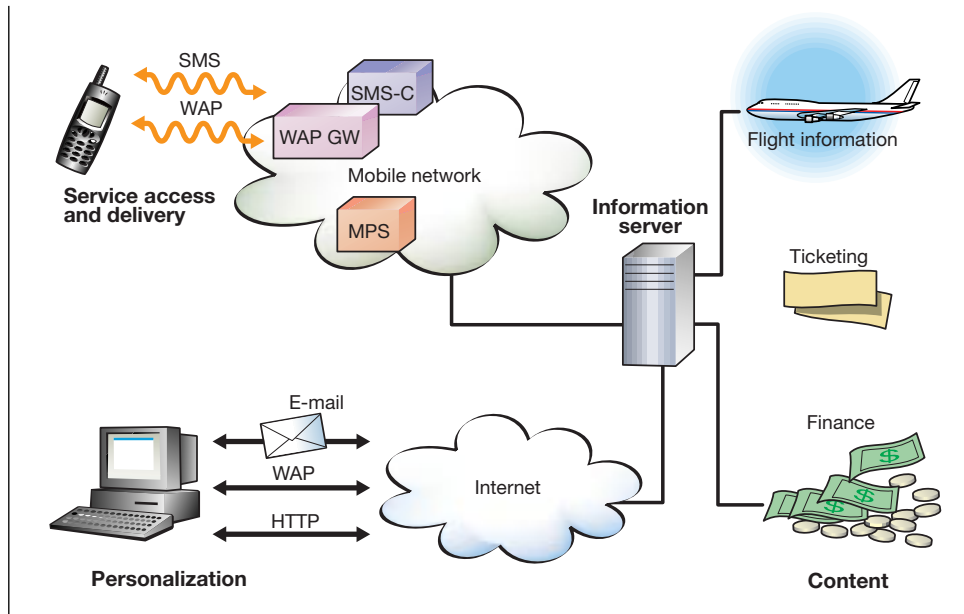
Technical overview

System domain

The WebOnAir Information Server serves as the connecting edge between mobile users, content providers and the mobile network. Its functions can be summarized as follows:

- it aggregates multiple-format, online information from different content providers into a pool of normalized data;
- it collects personal user preferences;
- it evaluates the normalized data according to user preferences and the information residing in the mobile network (position, for example) and distills important information at a given time or when the user requests it;

Figure 10
The WebOnAir Information Server system domain.



- it generates the reply to a user request in the desired format; and
- it delivers the generated information instantly to the user.

Basic concepts

Mobile service

A mobile service is implemented in the WebOnAir Information Server by two components: the feed handler (FH) and the service application (SA).

Feed handler

The feed handler is responsible for supplying the content provided by an external source (the content provider) to the service application in XML format. Nonetheless, a feed handler is not simply an adapter that

converts data from one format to another—in the WebOnAir Information Server, it is perceived as an intelligent component that can filter and deliver parts of available content in the external source on request or on a subscription basis.

A WebOnAir Information Server feed handler usually provides an interface, which allows the service application to subscribe to specific content categories. Feed handlers can serve content to one or more service applications by creating content channels for each one of them.

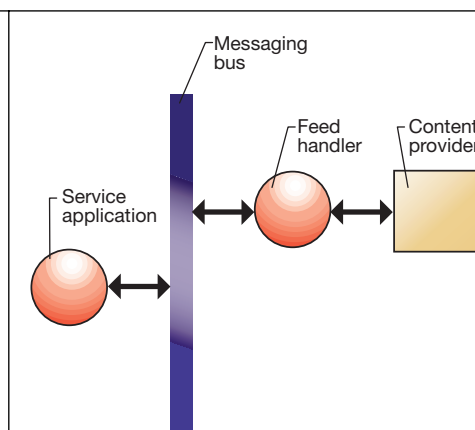
Conceptually, a feed handler implements logic, which allows it to function as a content proxy that presents a standardized XML interface to the service applications. This enables the WebOnAir Information Server to handle multiple-format content delivered by content providers and to work internally with one common format (XML).

Service application

The service application implements the behavior of a mobile service. Because service applications communicate with feed handlers using XML, they can be written once and run with any feed handler that provides the same interface. Like feed handlers, the service applications produce output that is formatted in XML.

Note: depending on the nature of the mobile service, the functions of the service application and feed handler components can coincide. In this case, the feed handler becomes a part of the service application.

Figure 11
The WebOnAir Information Server mobile service paradigm.



Messaging bus

Messaging bus software provides software components that can communicate across processes, nodes and networks in manner which is transparent to nodes and which uses a variety of communication paradigms, such as peer-to-peer, publish/subscribe, broadcast, and so on.

The core platform

Runtime kit

The runtime kit provides the infrastructure services necessary for deploying mobile services. The WebOnAir Information Server runtime kit includes the following components:

- The operation & maintenance subsystem (OMSS)—responsible for starting and stopping components, monitoring system activities, handling alarms and (if necessary) automatically restarting inactive components. The OMSS can forward alarms to any SNMP-compliant O&M system.
- The charging subsystem (CSS)—handles the storage of charging records generated by mobile services. This includes hierarchical storage and the automatic daily, monthly and yearly rollover of charging records.
- The Internet subsystem (ISS)—handles WAP and HTTP requests; that is, it dispatches them to the different mobile services and forwards service replies to the user.

- The transport subsystem (TPSS)—serves as the WebOnAir Information Server proxy to the GSM network and is responsible for handling SMS messages that originate and terminate in the mobile terminal (MT SMS).
- The top-level service application (TLSA)—responsible for authentication and subscriber management. The TLSA can use any LDAP v2/v3-compliant server to store subscriber data.

In addition, the following third-party components belong to the WebOnAir Information Server runtime kit:

- The Apache WebServer with hypertext preprocessing (PHP) extension—Apache provides access to the WebOnAir Information Server via HTTP and WAP; PHP is used for device-specific presentation services.
- The OpenLDAP Server—a freely available LDAP v2-compliant server.
- The Sybase Adaptive Server Enterprise is used for handling persistence.
- The TIB/Rendezvous messaging bus (TIB/RV) provides transparent communication facilities over a local area network (LAN) or wide-area network (WAN).

Figure 12 shows the components of the WebOnAir Information Server runtime kit. It also shows a generic feed handler and service application. Note that this is just one of any number of examples.

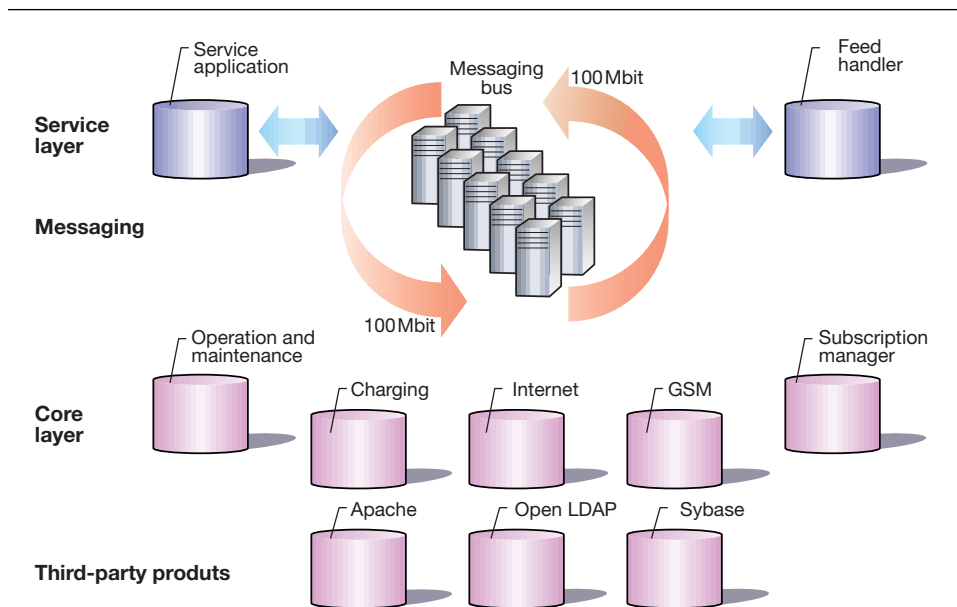


Figure 12
Component view of the WebOnAir Information Server runtime kit and services.

Applied scalability

Because a messaging bus is used, where the runtime kit and the service components are actually running is irrelevant. Figure 13 illustrates three configurations from a small to a mission-critical system.

Handling WAP requests

A user WAP/HTTP request is handled as follows:

1. The subscriber request arrives via HTTP to the WOAIS WebServer (Apache).
2. A WOAIS PHP library call is executed, which streams the request via a TCP/IP connection to the WOAIS Internet subsystem (ISS).
3. The ISS redirects the request via the internal communication subsystem (ICSS) to the corresponding service application (SA).
4. The SA replies by sending an XML response message to the ISS, which streams the response via a TCP/IP connection to the WOAIS PHP module. The response is returned in the library function mentioned in step 2.
5. The WOAIS PHP function optionally formats the XML response in WML or HTML and returns it to the user client.

Figure 14 illustrates the handling of a WAP request.

Processing short messages that originate from a terminal

A user originated request is handled as follows (Figure 15):

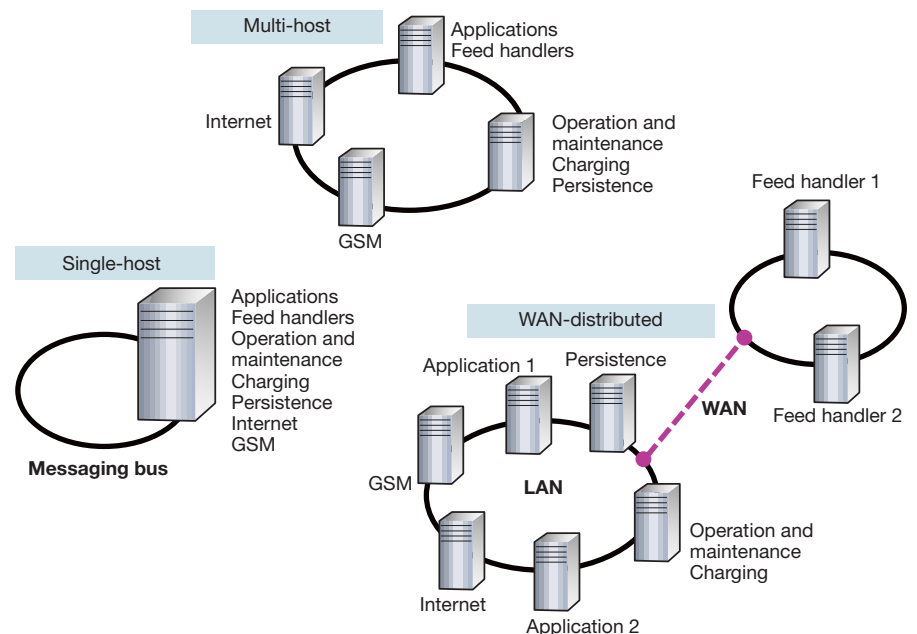
1. The user request arrives via the SMS center to the WOAIS SMS gateway (across wireless SMS transport gateway, AviSIM).
2. The AviSIM forwards the user requests to the WebOnAir Information Server transport subsystem (TPSS) via a TCP/IP connection.
3. The TPSS redirects the request via the internal communication subsystem to the corresponding service application.
4. The SA replies by sending an (internationalized) SMS response message to the TPSS.
5. The TPSS encodes the message and forwards it to AviSIM, which then sends it to the SMS-C.

Instead of being connected to an SMS-C, the TPSS can be connected to a user control point (UCP) server. Steps 1 to 5 remain the same. A short message (SMS) that terminates in a mobile terminal works in the same way as the reply to a short message (SMS) that originated in a mobile terminal.

The development kit

The development kit gives access to the entire functionality of the WebOnAir Infor-

Figure 13
Scaling via the messaging bus.



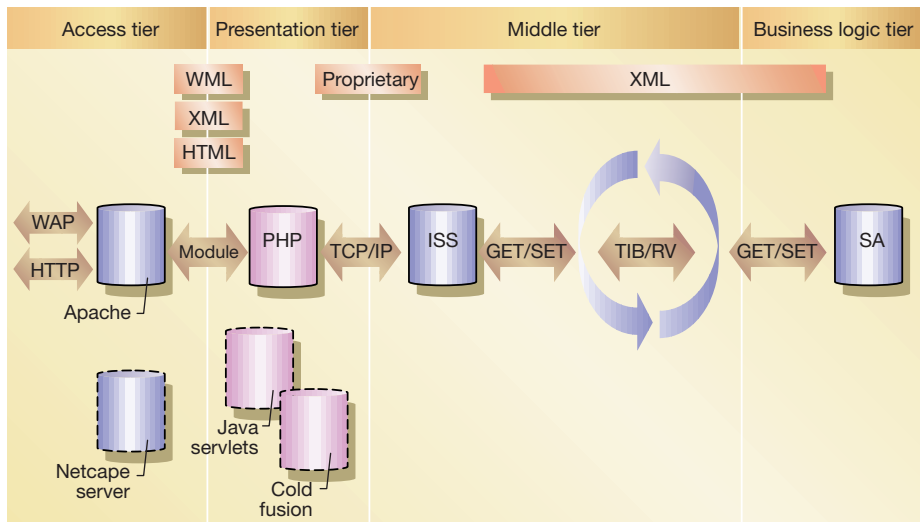


Figure 14
The handling of a WAP request.

mation Server, thus allowing the development of mobile services that fully exploit its strengths. The kit has been specifically designed for application service providers who want to achieve service differentiation and high scalability. The development kit consists of the following parts:

- The application development kit (ADK) provides complete information on the underlying concepts and techniques that can be used to develop highly personalized, situation-oriented, high-speed services.
- The software development kit (SDK) consists of the WebOnAir Information Server public application program interfaces

The application development kit comes with a fully functional template of a mobile service in source code, and with development guidelines for the rapid development of new mobile services. The kit also includes reference examples of generic service applications, and feed handlers that are ready for use as a basis for the developer's own applications.

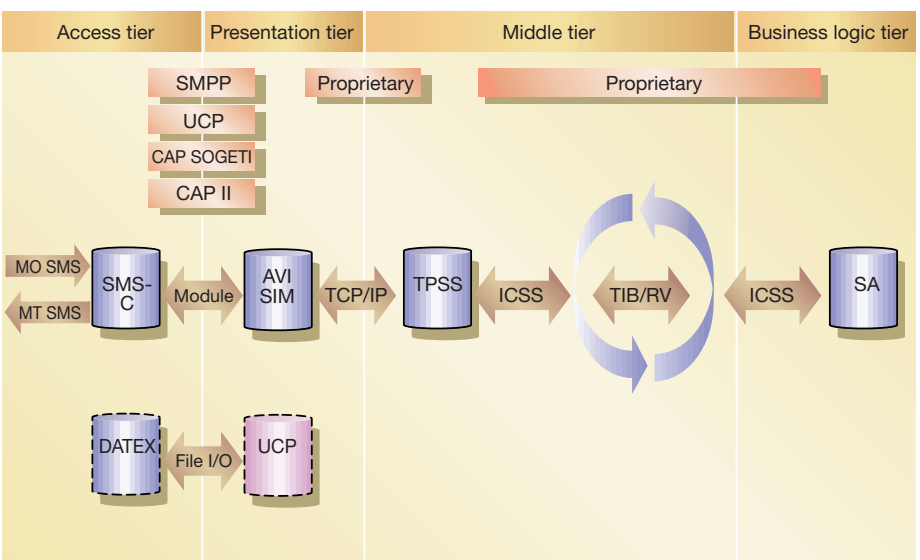


Figure 15
The handling of an SMS request originating in the mobile terminal (MO SMS).

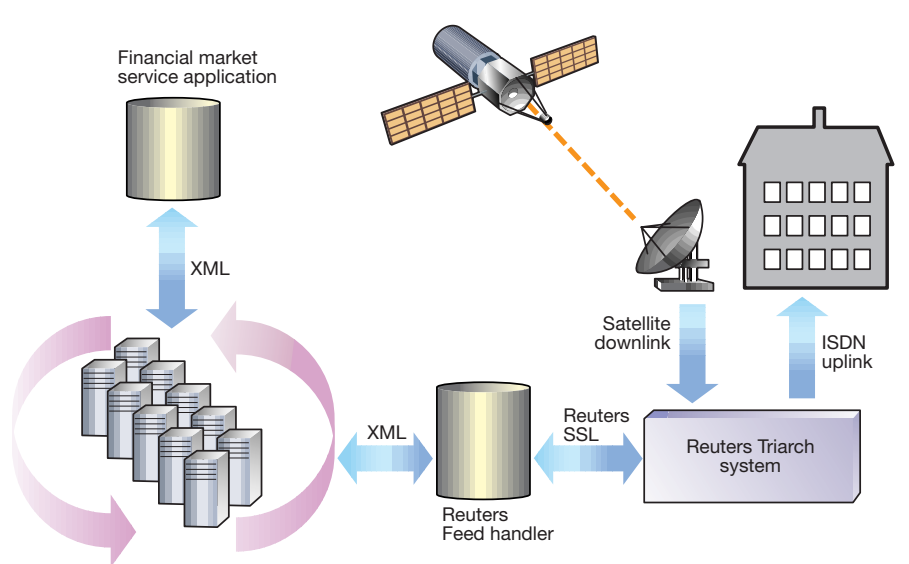


Figure 16
Realization of the Financial Market Data service.

Because of the high real-time demands, both the service application and feed handlers are built as multithreaded servers. Content provided by the Reuters feed is kept in memory for maximum performance. Financial Market Data offers

- the capacity to retrieve financial market information via SMS- or WAP-enabled devices;
- a personalized reporting service where users can set up their own portfolios. Users can request the current value of their stocks at any time using mobile-originated SMS or WAP;
- instant reporting when critical portfolio values are reached. Thresholds in absolute amounts or percentage value changes can be set for every stock in the portfolio. When a threshold is crossed, an SMS message is generated and sent to the user immediately; and
- easy tracking of standard values. The values of stocks in a portfolio can be sent automatically at specific times.

Conclusion

Mobile devices will become indispensable tools that enhance our daily lives. Applications and content revolve around the mobility, location and situation of the user. They become “situation-oriented.” This intelligence can be used to create highly valuable, personalized services that generate impressive revenues for operators. The challenge is to provide the right information at the right time.

Ericsson’s WebOnAir Information Server meets this challenge. It creates the infrastructure that allows rapid implementation and delivery of a new type of service; it enables developers to write applications that support all digital mobile phones on the market; and it offers time-based, event-based and on-demand services over SMS as well as the next generation of WAP phones.

In future releases, the WebOnAir Information Server will be integrated with mobile positioning, mobile e-payment, and WebOnAir interactive media.

(API), API documentation and examples of source code. Like the ADK, the SDK can be provided in source code.

On request, the development kit can be augmented by on-site training courses from experienced WebOnAir Information Server developers.

Applications

Financial Market Data

Financial Market Data—an industry-strength application based on WebOnAir Information Server technology—is a high-speed, real-time stock market service that can alert subscribers with SMS- or WAP-enabled devices to critical stock market events. The application is a real-time stock service built by means of the WebOnAir Information Server development kit. It is based on the WebOnAir Information Server service paradigm and uses a service application and a feed handler to the stock market feed from Reuters.

TRADEMARKS

WebOnAir is a trademark owned by Telefonaktiebolaget LM Ericsson.
OpenLDAP is a trademark of the OpenLDAP Foundation.
Sybase and Sybase Adaptive Server Enterprise are trademarks or registered trademarks of Sybase, Inc.
TIB/Rendezvous are trademarks or registered trademarks of TIBCO Finance Technology, Inc. and TIBCO Software, Inc.