[0065] The word "edge" is used for services forming interfaces to external networks or mobile network domains, the remainder being for internal processing within the gateway.

[0066] The manner in which an individual service is instantiated is illustrated in **FIG. 3**.

[0067] A service framework **41** comprises executable code for parsing an XML configuration document **40** and instantiating components of the target service according to parsing of the instructions. There may be more than one XML configuration document. **FIG. 3** shows generation of a CORBA protocol handler **42**, an XXX to MAMP component **43**, logic components **44** and **45**, and a MAMP to XXX component **46**. The components are instantiated by the framework **41** according to the parsed XML. The term "XXX" is used to highlight the versatile nature of the operation of the framework **41** for configuration. It may instantiate components for a variety of formats to/from MAMP, the internal language. The components register themselves automatically to the service framework **41** in an autonomous manner. The registration is indicated by the arrows from the components to the framework **41**. As part of instantiation of the components the framework **41** stores information for routing of messages internally within the service. This is also gleaned by parsing the XML document **40**.

[0068] Referring to **FIG. 4** an example service **50** comprises the following components:

[0069] **53**: CORBA protocol handler;

[0070] **54**: SMI to MAMP converter;

[0071] **55**: MAMP to HTML converter;

[0072] **56**, **57**: logic components for business rules; and

[0073] **58**: access control.

[0074] As shown in **FIG. 4**, the framework **41** also plays a very important role after configuration, in run-time. It comprises an event dispatcher **51** and a queue mechanism **52**, described in more detail below.

[0075] A service **100** shown in **FIG. 5** is an edge service, interfacing with external (of the gateway **1**) HTTP devices **102**. Its components include:

[0076] a HTTP protocol handler **101**, with MGS to HttpPush interaction functionality,

[0077] an SDL parser **103** containing model store **104**, translator **105**, and SMIL generator objects for converting content from external formats to SMIL format for delivery to mobile devices;

[0078] a MAMP to SDL converter **107**;

[0079] an SDL to MAMP converter **108**; and

[0080] a service framework **109** having internal components **110** including an event dispatcher.

[0081] It will thus be appreciated that the service framework allows a wide variety of services to be generated in a versatile manner according to XML instructions encoded in an XML configuration document. The precise mechanism for generating the components from XML is by the dynamic loading of modules or components referenced in the XML configuration document.

[0082] Internal communication within a service is illustrated in **FIG. 6** as a flow sequence of steps **121** to **127**. In step **121** a CORBA protocol handler ("CPH") receives an event in MAMP format from another service. In step **122** the CPH places an event in a queue of its service framework's queueing mechanism **52**. The service framework's event dispatcher **51** in steps **123** and **124** posts the event as an event object for a component of the service according to the stored internal routing information. This component processes the event in step **125** and posts a further event in a queue in step **126**. In step **127** the event is then dispatched to the next component of the service, again according to the routing information. Thus, internal communication within each service is achieved using the framework's queueing mechanism **52** as a hub for internal events, under control of the event dispatcher **51** using the routing information stored upon generation of the service from the XML configuration document.

[0083] **FIG. 7** illustrates inter-service communication within the gateway **1** as a method **140** having steps **141** to **150**. In step **141** an edge service receives a message. The protocol for content for messages within the gateway **1** is as proprietary internal XML-based language called MAMP. Irrespective of the protocol/format of incoming content it is always converted within the gateway to MAMP. In step **142** the message is posted onto a default channel in CORBA as it does not have a work flow routing list.

[0084] In step **143** a work flow manager ("WFM") service, which is listening on the default CORBA channel receives the message. In step **144** the WFM service determines an initial work flow list for the messages according to its logic component(s) and appends the list as MAMP to the message. The CPH component of the WFM service then reads the list and places the message on the relevant channel for CORBA **20**. Step **145** indicates overall routing of the message between services arising from the CPH of each service in turn reading the work flow list and placing the message on the relevant CORBA **20** channel. When it reaches a service without a next service indicator in the list the protocol handler (CPH) places the message on the default CORBA channel, from where it is retrieved by the WFM service in step **146**. This service does not "know" that it has previously processed this message, and it is simply treated as any other message retrieved from the default CORBA channel. Thus the WFM service parses the MAMP and according to its logic component(s) applies a work flow to the message in step **148**. Automatically, the message is routed in step **149** to an edge service, having an external MAMP to XXX component which routes the message externally of the gateway to the relevant enabler in the mobile network domain in step **150**.

[0085] It should be noted that the final destination of an incoming message need not necessarily be off the gateway **1**. It could alternatively be an internal service message.

[0086] Referring to **FIG. 8**, communication of content messages between the gateway **1** ("Hydra") and the external systems in the Internet and mobile network domains is illustrated. In this diagram, notation above the arrowed lines indicates the bearer (e.g. SMS) while notation below these