WAP Billing Framework Version 1.0

Version 21-Nov -2002

Open Mobile Alliance
OMA-WBF-v1_0-20021121-C

Continues the Technical Activities
Originated in the WAP Forum

This document is available online in PDF format at http://www.openmobilealliance.org/.

Known problems associated with this document are published at http://www.openmobilealliance.org/.

Comments regarding this document can be submitted to the Open Mobile Alliance™ in the manner published at http://www.openmobilealliance.org/technical.htm

Document History	
OMA-WBF-v1_0-20021121-C	Current
OMA-WBF-v1_0-20021121-C	Draft

Contents

1. Scope

Wireless Application Protocol (WAP), as part of Open Mobile Alliance (OMA), is a result of continuous work to define an industry wide specification for developing applications that operate over wireless communication networks. The scope for OMA is to define a set of specifications to be used by service applications. The wireless market is growing very quickly and reaching new customers and providing new services. To enable operators and manufacturers to meet the challenges in advanced services, differentiation, and fast/flexible service creation, WAP defines a set of protocols in transport, session and application layers. For additional information on the WAP architecture, refer to *"Wireless Application Protocol Architecture Specification"* [WAPArch].

The WAP Billing Framework specification defines the recording of service usage for *charging* purposes and its interfaces to the surrounding logical entities in the WAP Billing Framework architecture. It outlines possibilities to measure service usage by a specifically introduced concept called *chargeable operation*.
The recording can be performed in *realtime* (*Online Charging*) or *non realtime* (*Offline Charging*).
Note   The *realtime* process is not covered in this release of the specification.

Bills or direct payments towards for example credit cards or prepaid accounts may be issued as a result of recording service usage. However the WAP Billing Framework does not deal with the generation and presentation of bills or the actual payment and it does not specify specific Billing System features. For example an application performing *charging* against a pre-paid application is just as likely as one performing against a monthly bill or credit card.

It is the goal to define a framework that is flexible and extensible and that focuses on short time to market for deployment of a variety of billing models specifically based on measured service usage.

2. References

2.1. Normative References

[ClientID]   "WAP Client ID Specification", WAP Forum™. WAP-196-ClientID
                                        URL: http://www.openmobilealliance.org/

[CREQ]   "Specification of WAP Conformance Requirements". WAP Forum™. WAP-221-CREQ.
                                        URL:http//www.openmobilealliance.org/

[ISO4217]   "Codes for the representation of currencies and funds". URL: http://www.iso.ch

[PPGService]   "Push Proxy Gateway Service", WAP Forum™. WAP-249-PPGService
                                        URL: http://www.openmobilealliance.org/

[RFC1738]   "Uniform Resource Locators (URL)", T. Berners-Lee, et al., August 1998,
                                        URL:http://www.ietf.org/rfc/rfc1738.txt

[RFC2119]   "Key words for use in RFCs to Indicate Requirement Levels". S. Bradner. March 1997.
                                        URL:http://www.ietf.org/rfc/rfc2119.txt

[RFC2234]   "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell.
                                        November 1997. URL:http://www.ietf.org/rfc/rfc2234.txt

[RFC2387]   "The MIME Multipart/Related Content-type", E. Levinson, August 1998,
                                        URL:http://www.ietf.org/rfc/rfc2387.txt

[RFC2616]   "Hypertext Transfer Protocol -- HTTP/1.1", R. Fielding c.s., June 1999,
                                        URL:http://www.ietf.org/rfc/rfc2616.txt

[WAPPap]   "WAP Push Access Protocol Specification", WAP Forum™. WAP-247-PAPURL:
                                        http://www.openmobilealliance.org/

[WDP]   "Wireless Datagram Protocol", WAP Forum™. WAP-259-WDP,
                                        URL: http://www.openmobilealliance.org/

2.2. Informative References

[RFC640]   "Revised FTP Reply Codes ", Jon Postel, June 1975, URL:http://www.ietf.org/rfc/rfc640.txt

[WAPARCH]   "WAP Architecture". WAP Forum™. WAP-210-WAPArch.
                                        URL: http://www.openmobilealliance.org/

[WAPPush]   "WAP Push Architectural Overview", WAP Forum™. WAP-250-PushArchOverview
                                        URL: http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except "Scope" and "Introduction", are normative, unless they are explicitly indicated to be informative.

Defined terms (section 3.2) are written in *italic* in the following way:

> Party that utilizes the *charging* process in order to charge for the usage of his service.

3.2. Definitions

Chargeable operation
Defines a service delivery that has taken place in, and can be specified and recorded in, the WAP domain. *Chargeable operation* is used as a concept by which service delivery (e.g. WAP service usage) may be measured in the WAP domain. A *chargeable operation* is technically a sequence of events (request-response) in the context of the WAP architecture [WAPArch]. A set of *charging* data characterizes it and describes the service delivery that has taken place

Charged Party A party in the service execution which eventually gets charged for the service usage and with which the *charging party* has a service agreement, e.g. User, Push *content provider*, Advertiser.

Charging A function whereby information related to a *chargeable operation* is formatted and transferred in order to make it possible to determine usage for which the *charged party* may be billed.

Charging Data Provider
A logical entity in the WAP Billing Framework architecture. A *Charging Data Provider* is either of the components (*WAP) Proxy, Push Proxy Gateway* or *Content Provider*. A Charging Data Provider is an entity that is able to recognize and initiate the recording of a *chargeable operatio*n.

Charging Detail Record (CDR)
The CDR is a composition of *charging* data and additional administrative parameters. It may be all or a subset of the *charging* data that characterizes a recorded *chargeable operation.*

Charging Party Party that utilizes the *charging* process in order to charge for the usage of his service.

Content Provider (CP)
A logical entity in the WAP Billing Framework architecture which publishes content. The *Content Provider* may provide *charging* data to the Charging Control function. In such a case it is assumed that the *Content Provider* has a service agreement for Billing with the *charging party* in order to enable the *charging party* to charge on the CP's behalf.
Within WAP a *Content Provider* can typically be the Origin Server (OS) or the Push Initiator (PI).

Event based single recording
Recording of a complete *chargeable operation* using one single CDR.

Non realtime Any *charging* process that is not specifically defined as *realtime*. Typically this is the case when the service delivery is not dependent on the eventual *charging* for the usage.

Offline See *non realtime*. *Offline* is more commonly used when referring to the interface, with *non realtime* capabilities.

Online See *realtime*. *Online* is more commonly used when referring to the interface, with *realtime* capabilities.

Rating Determination of the total price of a resource usage (e.g. service) based on *charging* input and/or already priced input from the *Charging Data Provider.*

Realtime When the start of continuation of the service delivery to a user is dependent on the *charging* that is to be applied for the service usage or access, then this is generally being called a *realtime charging* process. Such dependencies exist for example in case of prepaid users where the account balance needs to be validated or in general to check the credit levels of users requesting a service.

This is sometimes also being referred to as pre-approval of a service execution based on the charges that will need to be applied.

Session based multiple recording

Recording of a *chargeable operation* by using more then one CDR. The first CDR will mark the start of the recording, while the last CDR will mark the stop of the recording. CDR's are uniquely numbered so that correlation of CDR's belonging to the same operation is possible. CDR's are unique based on the *Chargeable Operation* identification number together with the partial record sequence number and the identification of the *Charging Data Provider.*

Trusted access The interface/mechanism for data transfer between the Usage Recording function and a trusted party (i.e. a *Charging Data Provider*). A *trusted access* is implemented by a secure connection that does not allow manipulation of the data to be transferred. The implementation of this secure interface between the parties is beyond scope of the WAP Billing Framework specification.

3.3. Abbreviations

CCQ	Client Capability Query
CDR	*Charging Detail Record*
CP	*Content Provider*
GMT	Greenwich Mean Time
GPRS	General Packet Radio Service
HTTP	HyperText Transfer Protocol
ID	Identity
MIME	Multipurpose Internet Mail Extension
MSISDN	Mobile Subscriber ISDN
OMA	Open Mobile Alliance
OS	Origin Server
PI	Push Initiator
PPG	Push Proxy Gateway
RFC	Request For Comments
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UTC	Universal Co-ordinated Time
WAP	Wireless Application Protocol
WBF	WAP Billing FrameWork
XML	eXtensible Mark-up Language

4. Introduction

The *charging* criteria possible to use for services are fundamentally different from the ones traditionally used in telecom, such as merely measuring of connection time or data volume. These can of course still be used to charge on the bearer level and thereby indirectly charge end-users for services.

The WAP Billing Framework specification defines *chargeable operatio*ns for the purpose of measuring service usage. The information about *chargeable operations* is additional information to information recorded on the bearer level. Service usage is recorded on event (*event based single recording*) and/or transaction (*session based multiple recording*) basis and not on (call)session duration basis, which is typically associated with the events taking place on core network level (e.g. PDP context within GPRS).

Measured service usage opens up for a variety of new billing models. Not only the end-user is a presumed *charged party* but also application services, such as push initiators, can be charged for the services they use. Service usage measurement can be done in *realtime* (e.g. for prepaid purposes) or in *non realtime* using *online* and *offline* specified procedures respectively.
Note   The *realtime* process is not covered in this release of the specification.

The (WAP) proxy or push proxy gateway (PPG) can provide the operator (service provider) with sufficient amount of information on end-user or push initiator (PI) activities to generate *charging detail record*s (CDRs) for *charging* purposes. The *charging* data available in the (WAP) proxy and push proxy gateway can be recorded and used individually or in different combinations to generate revenue for pull and push services according to a variety of billing models.
In addition CDRs may be generated to support also other network and business management activities besides *charging*, e.g. interconnect settlement and statistical analysis.

The WAP architecture suggests that also the *Content Provider* can provide information that might be chargeable. There are various ways in which the *Content Provider* can provide chargeable information, i.e. directly from the content server or via the (WAP) proxy or push proxy gateway. It may depend on the business model applied by the Service Provider on what the most effective means of transport is in each case. However, the information will always be similar independent of which transfer mechanism is chosen.

With the introduction of the WAP2.0 specifications the usage of a proxy is no longer mandatory. In cases where service access and delivery is done without the help of a proxy, *charging* information will have to come from the *content provider*. Alternatively, *charging* the access for such services can be done on core network level provided that the recordings in the core network are capable of identifying the accessed service (which is out of scope for this specification).

The functions and data specified are operation and business oriented and are not in any aspect mandated in WAP. They are to be regarded as additional support functionality that may be motivated by a service provider's business model.

5. WAP Billing Framework Architecture   (Informative)

5.1. Architecture Overview

Figure 1 shows how the WAP Billing Framework architecture is mapped on a billing environment and how it extends from the WAP architecture. For the purpose of defining the WAP Billing Framework (WBF) the logical functional components called Usage Recording and Charging Control are introduced. Typically, this functionality can be included in a stand-alone system (for example as part of the Billing mediation layer) or can be distributed on the (WAP) Proxy, Push Pro xy Gateway or the *Content Provider*.

The Usage Recording function interfaces with existing components ([WAPArch] [WAPPush]) that measure the service usage and provide the Usage Recording function with information (i.e. *charging* data) about consumed resources. These components are also referred to here as *Charging Data Provider*s. The (WAP) proxy, Push Proxy Gateway and *Content Provider* are all possible *Charging Data Provider*s.
Once the Usage Recording function has collected the details of the service usage from the *Charging Data Provider*s it generates *charging detail record*s (CDR). In order to make the CDRs available for further processing the Usage Recording function interfaces with the Charging Control component in the Billing environment. The Usage Recording function is being used for *offline Charging* purposes.

The WAP Billing Framework provides an extensible architecture. New components, i.e. new services, new WBF functions or interfaces, may easily be added.
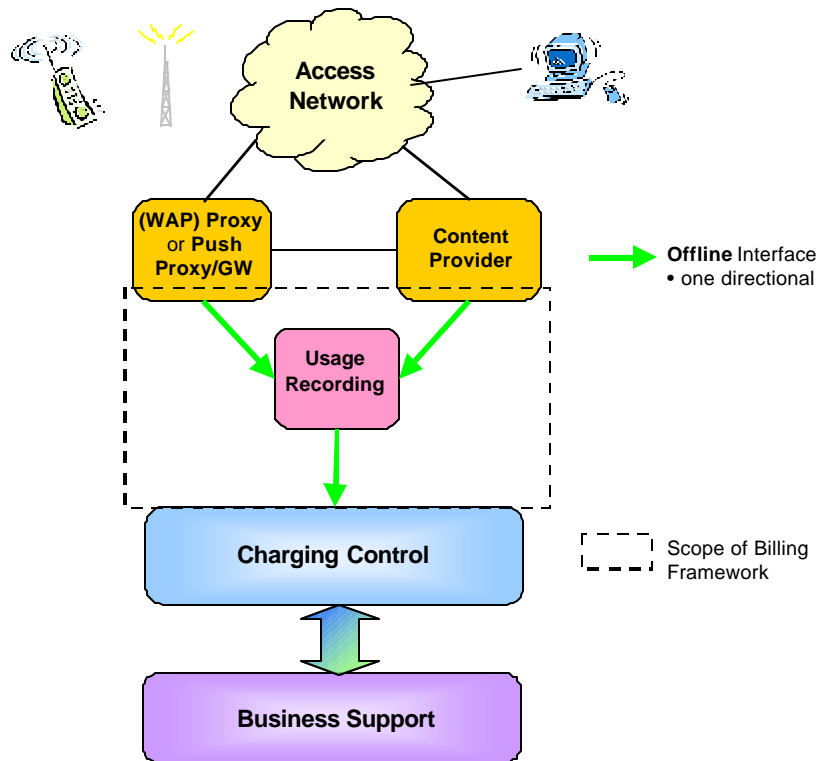

Figure 1. WAP Billing Framework Architecture

5.2. WAP Billing Framework Functions

5.2.1. Charging Data Providers

5.2.1.1. The (WAP) proxy

In case of WAP browsing (i.e. pull) the (WAP) proxy provides the resources and the service delivery (i.e. usage) that should be recorded. It operates in-between the bearer and application levels and can therefore meter usage of the individual pull attempts. It can provide details of the service usage for the *chargeable operation* called **Content pull**.

5.2.1.2. The Push Proxy Gateway

In case of push services the Push Proxy Gateway (PPG) is in the position to meter the push service usage. It interfaces to the Push Initiator by providing access from the internet side and can provide information of that usage. It interfaces the client by providing Message Handling Service and provides the usage details. For a description of the mentioned PPG functionality and interaction with the Push Initiators and clients refer to [WAPPush].

The following *chargeable operation*s that may be measurable by the PPG are defined in this specification: **Push submission**, **Push message delivery**, **Push Cancellation** and **Push Query**.

5.2.1.3. The Content Provider

In case *charging* data associated with a particular content are to be recorded (e.g. price/payment information, service details) such *charging* data may be made available by a *Content Provider*. This may be done directly from the *Content Provider* which is defined as the *chargeable operation **Content Provider Recording***.

Alternatively this type of recording can be done by including application level *charging* data in the HTTP-header together with the actual content, which can then be extracted by the Usage Recording function (via e.g. the Proxy) to be included in the recording of chargeable data. This feature is defined in this specification as the *chargeable operation **Recording of tag-based information***. A tag-based *charging* data transfer mechanism is provided in chapter 6.2.3.2.

This mechanism is in principle only applicable for *non realtime charging* procedures. This because for *realtime charging* procedures service delivery would already have been started when the data is extracted at the usage recording level.

The *Content Provider* may be responsible for setting the price of the content and include the resulting price in the information provided to Charging Control. Charging Control can then update the account accordingly and/or make sure that payment actually takes place. Alternatively, setting the price of the content can be outsourced to Charging Control in which case the *Content Provider* needs to include all the necessary data for correct *rating* of the content in the information stream provided to Charging Control.

5.2.2. Usage Recording

The service usage is measured in terms of *chargeable operations* that are specified in terms of descriptions and characterizing *charging* data. The Usage Recording function records the *chargeable operation*s by generating CDRs that contain the *charging* data. Each *chargeable operation* has one or more corresponding CDR's.

The Usage Recording function supports record generation of CDR's and provides mechanisms for the transfer of CDR's towards Charging Control.

Usage Recording as defined here is a logical entity, not a physical network element. However, it is a prerequisite that the Usage Recording function is allocated to a network element (or elements) that has *trusted access* to the *charging* data including the identity of the party to be charged and the identity of the service and/or content provider.

There may be one or more instances of the Usage Recording function resulting in one or more interfaces between Usage Recording and Charging Control. For example, in cases where the Usage Recording is distributed on the *Charging Data Provider*s each *Charging Data Provider* will have its own interface towards Charging Control.

5.2.3. Charging Control

Charging Control is the logical entity that processes the recorded chargeable data. The processing of chargeable data is out of scope for this specification.

Charging Control may be responsible for the *rating* of the content in order to determine the price that needs to be paid. In that case the necessary data to perform that *rating* needs to be included in the information stream from the *Content Provider* towards Charging Control. Alternatively, *rating* of the content (price determination) may be included at the *Content Provider*, after which the price is forwarded in the information provided to Charging Control for account update and/or payment.

Charging Control typically also acts as a mediator for the *charging* data it receives from various network elements for the same transaction. It may pre-process *charging* data (e.g. *rating*, reformatting), correlate data, filter data or enhance data based on implementation specific rules and features.
This role may be instead of or in addition to the mediator role possibly located on the Usage Recording function. The latter is very much implementation dependent.

5.2.4. Business Support

Business Support receives processed chargeable data from Charging Control. Here, the data is being transformed into Bills, processed for statistical purposes, prepared for wholesale settlement etc. using customer care & billing systems. These processes are out of scope for this specification.

6. Offline Charging

6.1. Charging Principles

The Usage Recording function provides recording of *chargeable operation*s. It permits composition of CDRs consisting of a different set of data items depending on the *chargeable operation* that has occurred.

The generated CDRs are transmitted to, or collected by an external processing system. In order to provide the Usage Recording functionality there are prerequisites and requirements that must be fulfilled. The prerequisites below outline the functionality that is required, however solutions for those are not covered by this specification. The requirements below outlines the features that are covered by this specification.

6.1.1. Prerequisites

- It is assumed that service delivery is measurable by the *Charging Data Provider*s.

- The *Charging Data Provider*s are assumed to be able to measure at least one of the *chargeable operation*s defined in this specification.

- It is assumed that parties involved in a operation have been authenticated prior to operation execution.

- It is assumed that in case of end-to-end secure connections the *Content Provider* will be able to record the *chargeable operation.*

- This specification does not cover how a client identity is forwarded to the *Charging Data Provider.*

- The identity of the party to be charged for the measured usage is assumed to be known by the Usage Recording function at the time of record generation. The Usage Recording function is assumed to have *trusted access* to *charging* data recorded by it.

- The management of the reception of different versions for the CDR in *Charging Control* is assumed to be a responsebility of *Charging Control*, which is out of scope for this specification.

6.1.2. Requirements

The requirements listed here are general for the specification. Detailed requirements for the Charging Data Collection and data transfer mechanisms are listed in the follow chapters.

- In case of a proxy-less access to content, recording of *chargeable operation*s MUST be done by the *Content Provider.* If a proxy is being used for access to content then recording of *chargeable operation*s MUST be done from the proxy or the *Content Provider*.

- Based on the provided *charging* data from the *Charging Data Provider*s, the Usage Recording function MUST support generation of CDR's.

- Each *Charging Data Provider* MUST generate its data for its own record types, i.e. the Proxy provides data for the *chargeable operation* **Content Pull**, the Push Proxy Gateway provides data for the *chargeable operation*s **Push Submission**, **Push Message Delivery**, **Push Cancellation** and **Push Query**, and the *Content Provider* provides data for the *chargeable operation* **Content provider recording**.
  The *Content Provider* MAY support the tagged-based *charging* transfer mechanism which enables the Proxy to extract pricing details allowing the Usage Recording function to create the combined **Pull Detail and Tag-based Information**.

6.2. Charging Data Collection

6.2.1. Notation Used within Charging Detail Record specification

A *charging detail record* (CDR) specifies a solution for how *charging* data and additional administrative data about a *chargeable operation* MUST be composed and presented.

A *charging detail record* is a record of service usage as defined by the corresponding *chargeable operation*. The *charging detail record*s define the *offline* interface to Charging Control on a logical level. (See figure 1).

Note the use of the term "optional" and "conditional" in the CDR status field.

The status field in the CDRs has the following meaning:

M	Mandatory. The parameter MUST be present in the record
O	Optional. The parameter MAY be present in the record.
	The parameter is only available under certain implementation specific conditions. The parameter(s) MAY be supported by the Usage Recording function. The product vendor may choose to include the parameter because the marketplace requires it or because the vendor feels it enhances the Usage Recording function.
C	Conditional. The parameter MAY be present in the record.
	The parameter is only available under certain specific conditions, not implementation related. If these conditions are met then the parameter MUST be supported by the Usage Recording function.

6.2.2. CDR Record

The CDR is made up of a combination of data items. Unless specifically specified each data item, if recorded, MUST only be recorded once in the CDR.
One CDR MUST include the recording for

- a complete *chargeable operation*, i.e. *event based single recording* of an operation

    or

- part of a *chargeable operation*, i.e. *session based multiple recording* where more then one CDR is used for the recording of a complete operation.

Independent of the *chargeable operation* all CDR's MUST include a specific set of data items which is specified in the table below:

Field	Status	Description	Field Type/Value
record-type	M	Identifies the type of *chargeable operation* that is recorded	One choice of "pull" or "push"
recording-entity	M	Identity of the entity where the Usage Recording function resides	IP-address
cdr-id	M	CDR record sequence number assigned by the recording identity. Value range MUST be from $0..2^{32}-1$	Numeric
chargeable-operation-id-number	M	Unique identification of the *Chargeable Operation* within the *Charging Data Provider*. Value range MUST be from $0..2^{32}-1$	Numeric

Field	Status	Description	Field Type/Value
timestamp	M	Time of recorded *chargeable operation*. The time-stamp indicates the time of completion of the operation. Completion is defined per *chargeable operation*. In case of recording of events in error situations the time-stamp marks the time of error recording. The contents of this field MUST be of the UTC Time format, containing local time plus an offset to GMT time, i.e. YYMMDDhhmmssShhmm, where YY = Year 00 to 99 MM = Month 01 to 12 DD = Day 01 to 31 hh = hour 00 to 23 mm = minute 00 to 59 ss = second 00 to 59 S = Sign = "+", "-" hh = hour 00 to 23 mm = minute 00 to 59	See description column for details.
additional-parameter	O	A set of network/vendor specific extensions to the record	Text

Depending on the data item "record-type", the additional set of data items is defined by either "pull" (chapter 6.2.3, describing the *chargeable operations* for Pull) or "push" (chapter 6.2.4, describing the *chargeable operations* for Push)

6.2.3. Chargeable operations, PULL

It MAY be necessary that more than one CDR is needed for the recording of a chargeable pull operation (*session based multiple recording*). In that case the CDR MUST be able to indicate the start and stop records and any required intermediate records that together make up the *chargeable operation*.
Start of the recording MAY be preceded by explicit user authorization for service execution (e.g. a purchase decision).
In case of *session based multiple recording* for the *chargeable operation* it MUST be possible to set trigger conditions for when intermediate recordings are to be generated. These trigger conditions MUST include (but are not limited to) the generation of a recording due to:

- Time (duration) limit

- Data volume limit

Other trigger conditions could, for example, include the generation of an intermediate record for special events during service execution, e.g. additional purchases, levels in a game or the reception/sending of specific signals. This is however judged to be implementation dependent and is no further specified within this specification.

In case of *session based multiple recording* the dynamic contents of the CDR (e.g. volume, time) MUST be related to the recording period only. Charging Control may be capable of aggregating the subsequent recordings for one *chargeable operation*, but may choose instead to rate and bill each intermediate recording separately.

Separate descriptions and requirements for the various *chargeable operations* related to Pull are listed in subsequent chapters. Independent of which Pull operation is recorded though all recordings related to Pull MUST include a specific set of data items, which is specified in the table below:

Field	Status	Description	Field Type/Value
pull-type	M	Identifies the type of *chargeable operation* related to Pull that is recorded	One choice of "pull-detail", "combined-pull" or "content-provider"
record-status	M	In case *session based multiple recording* is needed to record the whole *chargeable operation*, this field indicates whether the record is a start, intermediate or stop record. Otherwise (*event based single recording* for the whole *chargeable operation*) this field indicates a single event.	One choice of : - "start" - "stop" - "intermediate" - "single"
pull-client-id	M	Served terminal client identity as identified in the mobile access network, e.g. MSISDN, proxy user identity, global client id… (bearer dependent, billing model dependent)	Format according to [ClientID]
connection-type	M	Indicates the type of connection that was achieved. In case of recording at the proxy this field indicates the connection type between the proxy and the terminal. In case of recording at the *content provider* the field value will indicate the type of connection with the application server. In case the information can not be obtained the value "unknown" MUST be output.	One choice of "connection-oriented", "secure-connection-oriented", "connectionless", "secure-connectionless" or "unknown".
charging-data-provider	M	The network element identity of the *Charging Data Provider*.	IP-address
partial-record-sequence-number	C	In case of *session based multiple recording* for the *Chargeable Operation*, this field indicates the running sequence number within that recording for the complete *Chargeable Operation*. A start record MUST receive value 1.	Numeric

Depending on the data item "pull-type", the additional set of data items is defined by either "pull-detail" (chapter 6.2.3.1, describing the *chargeable operation content pull*), "combined-pull (chapter 6.2.3.2, describing the *chargeable operation Recording of tag-based information*) or "content-provider" (chapter 6.2.3.3, describing the *chargeable operation Content Provider Recording*)

6.2.3.1. Content Pull

6.2.3.1.1. Description

The *chargeable operation content pull* is applicable to browsing and a (WAP) proxy measures the service delivery. The services performed by the proxy are described in [WAPArch]. Recording can be done event based (*event based single recording*) or can be session based (*session based multiple recording*) where the pull transaction from request to end of download can be recorded.

6.2.3.1.2. Requirements

The *chargeable operation content pull* is assumed to be measurable by the proxy.

The proxy is assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *content pull* at completion of the operation.
In case of *event based single recording*, completion of the *chargeable operation* MUST be reached when the response is send from the proxy to the terminal.
In case of *session based multiple recording* for the event, completion of the operation for the start record MUST be reached when the pull request has been received by the proxy. For the stop record the same conditions MUST apply as in the *event based single recording* case, while for intermediate recording the conditions are described in chapter 6.2.3.

In case of situations where an error has occurred preventing completion of the operation, the proxy MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *content pull*. In such a case the proxy MUST be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.3.1.3. Content Pull data items for CDR

Field	Status	Description	Field Type/Value
destination	M	The URL from where the content was retrieved	URL [RFC1738]
content-type	M	Type of content	MIME [RFC2387]
bearer	M	Bearer service used	Name of the bearer according to Bearer Type in [WDP]
header-volume	M	Total header size as it is sent in the response expressed in number of bytes.	Numeric
data-volume	M	Size of the content delivered to the client expressed in number of bytes. This is defined as the size of the actual content as it is sent in the response.	Numeric
iresult	O	Indicating the protocol status of the content fetch (e.g. http-status code [RFC2616], ftp-status code [RFC640})	Text
wresult	M	Indicating the status of the response delivery to the handset (successful/failed). In case of unknown status, value 'unknown' MUST be output	One choice of: - "successful" - "failed" - "unknown"

6.2.3.2. Recording of Tag-based Information

6.2.3.2.1. Description

The *chargeable operation Recording of tag-based information* is applicable when the HTTP-header for content contains price/payment information provided by the *Content Provider* and may occur during the operation *content pull*.

This price information is extracted by the Proxy and send in addition to the total set of *charging* data for the *chargeable operation content pull to* Usage Recording, which can then generate the appropriate CDR.

6.2.3.2.2. Requirements

Both the Usage Recording function and the *Content Provider* MAY support the *chargeable operation* **Recording of tag-based information** by supporting the same tag-based *charging* data transfer mechanism.

The Usage Recording function is assumed to have *trusted access* to the *charging* data relevant for **Recording of tag-based information** as provided by the *Content Provider*.

In case this *chargeable operation* is supported, then:

- The mechanism for the transfer of tagged based data, specifying details about the content, as specified in chapter 6.2.3.2.3 MUST be supported by both the Usage Recording function and the *Content Provider*. Whether or not a tagged based header with charging data is supported and to be sent to the proxy has to be part of mutual agreements between the *content provider* and the service provider.
  In case the proxy receives a version of the tagged based header it does not support, the header MUST be discarded, service delivery and recording of the *chargeable operation* MUST be done according to **content pull** (as if no header was received) and appropriate operational alarms must be generated at the proxy.

- In case the *Content Provider* provides the price and currency information, the *Content Provider* MAY specify caching validity in the response. If it can not be guaranteed that caching is correctly provided, the chargeable content MUST be retrieved from the *Content Provider* at each request.

- The Usage Recording function MUST record the **Recording of tag-based information** operation by generating the CDR *Combined pull detail and tag-based information* at completion of the combined operation **content pull** including **Recording of tag-based information**.
  Completion of the *chargeable operation* MUST be reached when the response has been sent from the proxy to the terminal and when tagged based header information has been received from the *content provider.*
  In case of *session based multiple recording* for the event only intermediate and stop records can be generated for this operation. These conditions are described in chapter 6.2.3.

- In situations where an error has occurred preventing completion of the operation, the Usage Recording function MAY record the **Recording of tag-based information** operation. In such a case the Usage Recording function SHOULD record *charging* data indicating that result.

- The start record is always generated before the tagged-based information has been received and before it is even known whether tagged based information will be received at all. Until tagged based header information has been received by the proxy, a **Content Pull** MUST be generated under the conditions as specified in chapter 6.2.3.1. As soon as the header information is received a **Recording of Tag-based Information** MUST be generated for the next recording. Correlation will be possible based on *Chargeable Operation* Identification in combination with a Partial Record Sequence Number.In case the *Content Provider* provides the price and currency information, the *Content Provider* MAY specify caching validity in the response. If it can not be guaranteed that caching is correctly provided, the chargeable content MUST be retrieved from the *Content Provider* at each request.

6.2.3.2.3. Tag-based Charging Data Transfer Mechanism

Charging data MAY be transferred from the *content provider* to the proxy in order to make recording of this data at the proxy possible and not necessarily depend on this data being provided from the *content provider* directly. In that case the mechanism described here specifies how tag-based *charging* data MUST be transferred from a *Content Provider* (merchant) to the Usage Recording function. The mechanism specified in this chapter is described in both prose and augmented Backus-Naur Form, refer to [RFC2234].

6.2.3.2.3.1. Requirements

Payment/Pricing information for specific content is encoded within the HTTP header (refer to [RFC2616]) and sent with the HTTP response from the *Content Provider*.

6.2.3.2.3.2. Format of HTTP Header Information

X- Payment-Info   = "X- Payment-Info" ":" charging-data

charging-data	= (	charging-data-header-version
		"," merchant-id
		"," pricing-info
		["," service-user-id]
		["," charged-party]
		["," transaction-id]
		["," description]
		["," additional])

charging-data-header-version	= "charging-data-header-version" "=" "oma-wbf-v1_0"
merchant-id	= "merchant-id" "=" 1*255 (ALPHA/DIGIT)
pricing-info	= (price	content-value-class	(price "," content-value-class))
price	= (amount "," currency)
amount	= "price" "=" ["-"]1*10 DIGIT
	;(no decimals, digits according to [ISO4217])
currency	= "currency" "=" 3 ALPHA
	;(e.g. EUR, letters according to [ISO4217])
content-value-class	="content-value-class" "=" 1*10 DIGIT
service-user-id	= "service-user-id" "=" 1*30 (ALPHA/DIGIT)
charged-party	= "charged-party" "=" 1*30 (ALPHA/DIGIT)
transaction-id	= "transaction-id" "=" 1*30 (ALPHA/DIGIT)
description	= "description" "=" 1*30 TEXT ; comma is not allowed here,
additional	= "additional" "=" 1*128 TEXT

All fields are delimited by comma (","). The ALPHA, DIGIT and TEXT rules are defined in [RFC2616].

charging-data-header-version   This field is REQUIRED and has a fixed value of "oma-wbf-v1_0" to state compliance to the *charging* data header specified in this document.

merchant-id   String with maximum 255 characters, variable length as required. The merchant-id MUST be agreed between merchant and *charging party* and serves as an identifier for the merchant for billing purposes (e.g. a URI). The *charging party* MAY use this information to collect money for the merchant and to reimburse the merchant later. This field is REQUIRED.

pricing-info   is either the **price** field or the **content-value-class** field or both. This field is REQUIRED.

price   is made up of two REQUIRED fields, **amount** and **currenc**y. If price is not available, content-value class is mandatory.

amount   an optional minus sign followed by one to ten digits without decimal point. The number of decimals are defined in [ISO 4217:1995] for each currency. A positive price value (without minus) indicates that the value

should be debited from the *charged party*. Negative values are allowed for crediting the (normally) *charged party*. This field is conditionally required if the **price** field is present.

currency   three letters indicating the currency. Each possible currency is listed in [ISO4217]. This field is conditionally required if the **price** field is present.

content-value-class   Identifier for the value class of the content delivered to the user. (e.g. content may be classified into to a number of categories: free content, minimum rate content, premium rate content) The range and specific value have to be subject of mutual agreements between the service provider and the *Content Provider*. The service provider MAY use it as input to the *rating* for the content. This field is OPTIONAL. If content-value-class is not available, price is mandatory.

service-user-id   application level identity of the user who is requesting the respective page or using the application service of the merchant ([ClientID] or an implementation dependent string with maximum length of 30 characters). This field is OPTIONAL.

charged-party   OPTIONAL field, indicating a possible 3rd party that needs to be charged (debit/credit) for the service delivery (e.g. advertisers, sponsor). Business level agreements have to be in place between service and *content provider* to allow for this information to be provided and to agree on the accepted charged parties.

transaction-id   string with maximum length of 30 characters, The merchant MUST define a unique identifier of the respective transaction for reference reasons. This field is REQUIRED.

description   textual description of the tag-based payment information provided in the header (e.g. of the page requested). This should be used to describe the service where available and appropriate. String with maximum length of 30 characters. Commas MUST NOT be used within this field. This field is OPTIONAL.

additional   information not defined in this specification. String with maximum length of 128 characters. This field is OPTIONAL.

6.2.3.2.3.3. Example

```
X- Payment-Info:                    charging-data-version-header=oma-wbf-v1_0,

                                    merchant-id=A3F745CDD,

                                    price=2538,

                                    currency=EUR,

                                    service-user-id=386E,

                                    transaction-id=F77,

                                    description=Stock-info:Siemens
```

The header is compliant the header version oma-wbf-v1_0

The merchant identity is A3F745CDD.

The price is 25,38 with currency `EUR´.

The service-user-id is 386E.

The transaction-id as provided by the merchant is F77.

The description of the transaction is Stock-info:Siemens.

No additional information is given. No 3rd party is to be charged for the service.

6.2.3.2.4. Combined Pull Detail and Tag-based Information data items for CDR

Field	Status	Description	Field Type/Value
destination	M	The URL from where the content was retrieved	URL [RFC1738]
content-type	M	Type of content.	MIME [RFC2387]
bearer	M	Bearer service used	Name of the bearer according to Bearer Type in [WDP]
header-volume	M	Total header size as it is sent in the response expressed in number of bytes.	Numeric
data-volume	M	Size of the content delivered to the client expressed in number of bytes. This is defined as the size of the actual content as it is sent in the response.	Numeric
merchant-id	M	An identifier that identifies the *Content Provider* to the *charging party*.	ALPHA/DIGIT
iresult	O	Indicating the protocol status of the content fetch (e.g. http-status code [RFC2616], ftp-status code [RFC640})	Text
wresult	M	Indicating the status of the response delivery to the handset (successful/failed). In case of unknown status value 'unknown' MUST be output	One choice of: - "successful" - "failed" - "unknown"
content-value-class	C	Type of content delivered to the user, used as input for *rating* within Charging Control Output if the Price AND Currency are **not** used	MIME [RFC2387]
price	C	Amount to be charged for the service, both positive and negative amounts are allowed. Output if Content Value Class is **not** used.	DIGIT
currency	C	Three letter indication for the Currency being used Output if Content Value Class is **not** used.	ALPHA [ISO4217]
service-user-id	O	A merchant provided user id. For example a client identity or a *charging party* or merchant defined user name.	Format according to [ClientID] or implementation dependent string
charged-party	O	This field indicates a possible 3rd party that needs to be charged for the service	ALPHA/DIGIT
transaction-id	M	A merchant defined unique transaction identifier	ALPHA/DIGIT
descriptive-text	O	A merchant defined text string for informational purposes. This should be used to describe the service	Text

Field	Status	Description	Field Type/Value
		where available and appropriate.	

6.2.3.3. Content Provider Recording

6.2.3.3.1. Description

The *chargeable operation Content Provider Recording* is applicable to browsing where the *Content Provider* measures the service delivery. Recording can be done event based (*event based single recording*) or can be session based (*session based multiple recording*) where the pull transaction from request to end of download can be recorded.

This as an alternative to the *chargeable operation Content Pull* and *Recording of Tag-based information* where it is the proxy that takes care of the measurement.

For various purposes it may even be so that both the proxy and the *Content Provider* generate the CDR's specified here (validation, statistics, settlements etc.).

6.2.3.3.2. Requirements

The *chargeable operation Content Provider Recording* MUST be measurable by the *Content Provider*.

The *Content Provider* is assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Content Provider Recording* at completion of the operation.
 Completion of the *chargeable operation* MUST be reached when the response is sent from the *Content Provider*.
In case of *session based multiple recording* for the event, completion of the operation for the start record MUST be reached when the pull request has been received by the *content provider*. For the stop record the same conditions MUST apply as in the *event based single recording* case, while for intermediate recording the conditions are described in chapter 6.2.3.

The *Content Provider* MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Content Provider Recording* also in situations where an error has occurred preventing completion of the operation. In such a case the *Content Provider* SHOULD be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.3.3.3. Content Provider Recording data items for CDR

Field	Status	Description	Field Type/value
service-user-id	M	Application level identity of the user who is requesting the respective page or using the application service of the merchant.	Format according to [ClientID] or implementation dependent string.
charged-party	O	This field indicates a possible 3rd party that needs to be charged for the service	ALPHA/DIGIT
destination	M	The URL from where the content was retrieved	URL [RFC1738]
header-volume	M	Total header size as it is sent in the response expressed in number of bytes.	Numeric
data-volume	M	Size of the content delivered to the client expressed in number of bytes. This is defined as the size of the actual content as it is sent in the response.	Numeric
merchant-id	M	An identifier that identifies the *Content Provider* to	ALPHA/DIGIT

Field	Status	Description	Field Type/value
		the *charging party*. The merchant-id must be agreed between merchant and *charging party* and serves as an identifier for the merchant for billing purposes. The *charging party* MAY use this information to collect money for the merchant and to reimburse the merchant later.	
iresult	O	Indicating the protocol status of the content fetch (e.g. http-status code [RFC2616], ftp-status code [RFC640})	Text
wresult	M	Indicating the status of the response delivery to the handset (successful/failed). In case of unknown status value 'unknown' MUST be output	One choice of: - "successful" - "failed" - "unknown"
content-value-class	C	Identifier for the value class of the content delivered to the user agent. (e.g. content may be classified into to a number of categories: free content, minimum rate content, premium rate content) The range and specific value must be subject of mutual agreements between the service provider and the *Content Provider*. The service provider may use it as input to the *rating* for the content. Only output if the Price AND Currency are **not** used	MIME [RFC2387]
price	C	Amount to be charged for the service, both positive and negative amounts are allowed. In case price is output the field currency MUST be output as well.Only output if Content Value Class is **not** used.	DIGIT
currency	C	Three letter indication for the Currency being usedOnly output if Content Value Class is **not** used.	ALPHA [ISO4217]
transaction-id	M	A merchant defined unique transaction identifier	ALPHA/DIGIT
descriptive-text	O	A merchant defined text string for informational purposes. This should be used to describe the service where available and appropriate.	Text

6.2.4. Chargeable operations, PUSH

The descriptions and requirements for the various *chargeable operations* related to Push are listed in subsequent chapters. Independent of which Push operation is recorded though, all recordings related to Push events MUST include a specific set of data items, which is specified in the table below:

Field	Status	Description	Field Type/value
push-type	M	Identifies the type of *chargeable operation* related to Push that is recorded	One choice of "push-submission", "push-message", "push-

Field	Status	Description	Field Type/value
			cancellation" or "push-query".
pi-id	M	The identity of the PI.	IP-address
ppg-id	M	The identity of the PPG.	IP-address
push-id	M	A push message identity provided by the PI	Numeric

Depending on the data item "push-type", the additional set of data items is defined by either "push-submission" (chapter 6.2.4.1, describing the *chargeable operation Push Submission*), "push-message" (chapter 6.2.4.2, describing the *chargeable operation Push Message*), "push-cancellation" (chapter 6.2.4.3, describing the *chargeable operation Push Cancellation*) or "push-query" (chapter 6.2.4.4, describing the *chargeable operation Push Query*)

6.2.4.1. Push Submission

6.2.4.1.1. Description

The *chargeable operation Push submission* is applicable to the situation where the PPG interacts with a Push Initiator and the PI initiates a message delivery request or the PPG receives a message delivery initiation or replacement request, as described in [WAPPush].

6.2.4.1.2. Requirements

The *chargeable operation Push submission* MUST be measurable by either the PPG or the PI.

The PPG or the PI are assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Push submission* at completion of the operation. Completion is achieved by the PPG when the submission response has been sent and by the PI when the submission response has been received.

In situations where an error has occurred preventing completion of the *Push submission* operation, the PPG or the PI MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Push submission*. In such a case the PPG or the PI SHOULD be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.4.1.3. Push submission data items for CDR

Field	Status	Description	Field Type/value
replace-push-id	C	In case replacement of a previously submitted push is requested, then the same Push ID is repeated here to indicate that request.	Numeric
push-content-length	M	Push message data volume. This is defined as the entire message delivered from PI.	Numeric
push-content-type	M	Type of content	MIME [RFC2387]
priority	M	Delivery feature: push message handling priority	One choice of "high", "medium" or "low".
number-of-recipients	C	The number of recipients concerned with the *message*	Numeric

Field	Status	Description	Field Type/value
		delivery initiation of the push message. Only to be output if more then 1 recipient is addressed.	
status-code	O	The status of the message immediately after submission as defined in [WAPPap].	Numeric (four digits)

6.2.4.2. Push Message Delivery

6.2.4.2.1. Description

The *chargeable operation **Push message delivery*** is applicable to the situation where a PPG delivers push messages to clients as described in [WAPPush]. It is also applicable in situations where a PI receives a Result Notification from the PPG on push messages delivered to clients as described in [WAPPush].

Completion of the operation is dependent on the message delivery features leveraged by the PPG for a particular operation, as facilitated by the *charging* data relevant for ***Push message deliver*y**.

6.2.4.2.2. Requirements

The *chargeable operation **Push message delivery*** MUST be measurable by the PPG and MAY be measurable by the PI depending on whether the PI has requested for a result notification at submission of the push message.

The PPG or the PI are assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for ***Push message delivery*** at completion of the operation. At confirmed push, completion MUST be achieved by the PPG at Result Notification acknowledgement from the PI and at the PI at Result Notification acknowledgement sending. In case of unconfirmed push, completion of the operation MUST be rached at the PPG when the push message is delivered by the PPG.

In situations where an error has occurred preventing completion of the ***Push message delivery*** operation, the PPG or the PI MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for ***Push message delivery***. In such a case the PPG SHOULD be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.4.2.3. Push message delivery data items for CDR

Field	Status	Description	Field Type/value
push-client-id	C	Served terminal client identity as identified in the mobile access network, e.g. MSISDN, proxy user identity, global client id… (bearer dependent, billing model dependent). Only to be output if the chargeable data are generated by the PPG.	[ClientID]
recipient-address	M	Client address of the destination.	Client addressing according to [PPGService]
delivery-result	M	Delivery type, one of: • Unconfirmed Push Initiated	One choice of "unconfirmed-pi", "confirmed-push-success"or

Field	Status	Description	Field Type/value
		• Confirmed Push Success • Confirmed Push Failure	"confirmed-push-failure"
bearer	O	Beare r service utilised for delivery, if available	Text
message-state	O	Status of the message delivery as defined in [WAPPap]	One choice of "rejected", "pending", "delivered", "undeliverable", "expired", "aborted", "timeout", "cancelled" or "unknown"

6.2.4.3. Push Cancellation

6.2.4.3.1. Description

The *chargeable operation Push cancellation* is applicable to the situation where a PPG interacts with a push initiator (PI) for delivery cancellation as described in [WAPPush].

Completion of the operation is when the PPG or the PI has performed the delivery cancellation.

6.2.4.3.2. Requirements

The *chargeable operation Push cancellation* MUST be measurable by the PPG or the PI.

The PPG or the PI are assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Push cancellation* at completion of the operation. Completion is achieved at the PPG at acknowledging the cancellation and at the PI at reception of the acknowledgement.

In situations where an error has occurred preventing completion of the *Push cancellation* operation, the PPG or the PI MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for *Push cancellation*. In such a case the PPG SHOULD be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.4.3.3. Push cancellation data items for CDR

Field	Status	Description	Field Type/value
-	-	-	-

6.2.4.4. Push Query

6.2.4.4.1. Description

The *chargeable operation Push query* is applicable to the situation where a PPG interacts with a push initiator (PI) and manages queries as described in [WAPPush], i.e. Push Client Capability Query, Push Status Query and any other possible Push Queries that may be defined in the future.

Completion of the operation is when the PPG has responded to a query is when the PI has received the result to a query.

6.2.4.4.2. Requirements

The *chargeable operation **Push query*** MUST be measurable by the PPG or PI.

The PPG or PI are assumed to have *trusted access* to *charging* data and MUST be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for ***Push query*** at completion of the operation. Completion is achieved at the PPG at sending of the query result and at the PI at reception of the query result.

In situations where an error has occurred preventing completion of the ***Push query*** operation, the PPG or PI MAY be able to initiate record generation by providing the Usage Recording function with the *charging* data relevant for ***Push query.*** In such a case the PPG SHOULD be able to provide the Usage Recording function with *charging* data indicating that result.

6.2.4.4.3. Push query data items for CDR

Field	Status	Description	Field Type
response-code	O	Applicable for CCQ: *Successful query* or *Query denied*	Text

6.3. Charging Detail Record

6.3.1. Charging Data Provider – Usage Recording

This will not be specified, because it is considered to be implementation specific.

6.3.2. Usage Recording – Charging Control

6.3.2.1. CDR Media Type

The MIME media type for objects conforming to the format defined in this section MUST be

```
application/vnd.oma.wbf.cdr
```

6.3.2.2. CDR format

```
<?xml version="1.0" encoding="UTF-8"?>

<!--
Charging Detail Record (CDR) oma-wbf-v1_0 Document Type Definition.
Copyright Open Mobile Alliance 2002-2003
All rights reserved.
CDR is an XML language. Typical usage:

          <?xml version="1.0"?>
          <!DOCTYPE cdr PUBLIC "-//openmobilealliance.org//DTD oma-wbf-v1_0//EN"
          "http://www.openmobilealliance.org/DTD/oma-wbf-v1_0.dtd"
          <cdr>
                    ...
          </cdr>

Terms and conditions of use are available from the openmobilealliance.
web site at
http://www.openmobilealliance.org/omacopyrightNEW.asp
-->
```

```
<!--Definition of the Charging Detail Record -->

<!ELEMENT cdr (record-type, recording-entity, cdr-id, chargeable-operation-id-number, timestamp,
additional-parameter?)>
<!ELEMENT record-type (pull | push)>
<!ELEMENT recording-entity (#PCDATA)>
<!ELEMENT cdr-id (#PCDATA)>
<!ELEMENT chargeable-operation-id-number (#PCDATA)>
<!ELEMENT timestamp (#PCDATA)>
<!ELEMENT additional-parameter (#PCDATA)>
<!ELEMENT pull (pull-type, record-status, pull-client-id, connection-type, charging-data-
provider, partial-record-sequence-number?)>
<!ELEMENT pull-type (pull-detail | combined-pull | content-provider)>
<!ELEMENT record-status (#PCDATA)>
<!ATTLIST record-status
                status (start | stop | intermediate | single) #FIXED "single"
>
<!ELEMENT pull-client-id (#PCDATA)>
<!ELEMENT connection-type (#PCDATA)>
<!ATTLIST connection-type
                type (connection-oriented | secure-connection-oriented | connectionless | secure-
connectionless | unknown) #FIXED "unknown"
>
<!ELEMENT charging-data-provider (#PCDATA)>

<!--The partial-record-sequence-number is a conditional
                element In case of multiple recordings for the Chargeable
                Operation, this field indicates the sequence number within
                that recording for the complete Chargeable Operation.
                A start record MUST receive value 1.-->

<!ELEMENT partial-record-sequence-number (#PCDATA)>

<!--The pull-detail which is retrieved during conventional browsing.
                This data is  provided by the proxy.-->

<!ELEMENT pull-detail (destination, content-type, bearer, header-volume, data-volume, iresult?,
wresult)>
<!ELEMENT destination (#PCDATA)>
<!ELEMENT bearer (#PCDATA)>
<!ELEMENT header-volume (#PCDATA)>
<!ELEMENT data-volume (#PCDATA)>
<!ELEMENT iresult (#PCDATA)>
<!ELEMENT wresult (#PCDATA)>
<!ATTLIST wresult
                is (successful | failed | unknown) #FIXED "unknown"
>

<!--The combined-pull detail. This data are obtained by combining the billing,
                http headers from the origin server with the additional data from the proxy.-->

<!ELEMENT combined-pull (destination, content-type, bearer, header-volume, data-volume,
merchant-id, iresult?, wresult, content-value-class?, price?, currency?, service-user-id?,
charged-party?, transaction-id, descriptive-text?)>

<!--The content-type is a conditional element in.
                only to be output when content-value-class is not being used.-->

<!ELEMENT content-type (#PCDATA)>
<!ELEMENT merchant-id (#PCDATA)>

<!--The content-value-class is a conditional element,
                indicating ype of content delivered to the user,
                used as input for rating within Charging Control Output
```

```
                    if the Price AND Currency are not used-->

<!ELEMENT content-value-class (#PCDATA)>

<!--The price is a conditional element, amount to be charged for the service,
                 both positive and negative amounts are allowed.
                 Output if content-value-class is not used.-->

<!ELEMENT price (#PCDATA)>

<!--The currency is a conditional element, i.e. type of currency,
                 air-time minutes, "WAP-units" Ouput if Content Value Class is not used.-->

<!ELEMENT currency (#PCDATA)>
<!ELEMENT service-user-id (#PCDATA)>
<!ELEMENT charged-party (#PCDATA)>
<!ELEMENT transaction-id (#PCDATA)>
<!ELEMENT descriptive-text (#PCDATA)>

<!--pull-content-type, content-value-class, price, currency are conditional elements,
                 please refer to corresponding tables in specification for details.-->

<!ELEMENT content-provider (service-user-id, charged-party?, destination, header-volume, data-
volume, merchant-id, iresult?, wresult, content-value-class?, price?, currency?, transaction-id,
descriptive-text?)>
<!ELEMENT push (push-type, pi-id, ppg-id, push-id)>

<!--replace-push-id, number-of-recepients, client-id, service-user-id, pi-id,
                 ppg-id  are conditional records, please refer to corresponding
                 tables in specification for details.-->

<!ELEMENT push-type (push-submission | push-message-delivery | push-cancellation | push-query)>
<!ELEMENT pi-id (#PCDATA)>
<!ELEMENT ppg-id (#PCDATA)>
<!ELEMENT push-id (#PCDATA)>
<!ELEMENT push-submission (replace-push-id?, push-content-length, push-content-type, priority,
number-of-recipients?, status-code?)>

<!--The replace-push-id is a conditional element,
                 In case replacement of a previously submitted push is requested,
                 then the same Push ID is in this element to indicate that request.-->

<!ELEMENT replace-push-id (#PCDATA)>
<!ELEMENT push-content-length (#PCDATA)>
<!ELEMENT push-content-type (#PCDATA)>
<!ELEMENT priority (#PCDATA)>
<!ATTLIST priority
                 priority (high | medium | low) #FIXED "medium"
>

<!--The number-of-recipients is a conditional element,
                 The number of recipients concerned with the message delivery initiation of
                 the push message. Only to be output if more then 1 recipient is addressed.-->

<!ELEMENT number-of-recipients (#PCDATA)>
<!ELEMENT status-code (#PCDATA)>
<!ELEMENT push-message-delivery (push-client-id?, recipient-address, delivery-result, bearer?,
message-state?)>

<!--The push-client-id is a conditional element,
                 Served WAP terminal client identity as identified in the mobile access network,
                 e.g. MSISDN, WAP proxy user identity, global WAP client id
                 (bearer dependent, billing model dependent).
                 Only to be output if the chargeable data are generated by the PPG.-->
```

```
<!ELEMENT push-client-id (#PCDATA)>
<!ELEMENT recipient-address (#PCDATA)>
<!ELEMENT delivery-result (#PCDATA)>
<!ATTLIST delivery-result
                type (unconfirmed-pi | confirmed-push-success | confirmed-push-failure) #REQUIRED
>
<!ELEMENT message-state (#PCDATA)>
<!ATTLIST message-state
                status (rejected | pending | delivered | undeliverable | expired | aborted |
timeout | cancelled | unknown) #FIXED "unknown"
>

<!--The push-cancellation element has no value.
                If present, it is used to mark the cdr record as a push cancellation.-->

<!ELEMENT push-cancellation (#PCDATA)>
<!ELEMENT push-query (response-code?)>
<!ELEMENT response-code (#PCDATA)>
```

6.3.2.3. Sending of CDR's to Charging Control

CDR's can be forwarded to Charging Control one by one as they are generated or they can be collected in, for example, files and sent to Charging Control in intervals. Alternatively, Charging Control can poll the CDR's from the Usage Recording.

This will not be further specified in this version of the framework.

7. Online Charging (Informative)

The procedures for *online Charging*, i.e. *realtime charging* for content or the option to manage payments *online*, will be covered in a later release of this specification.

Appendix A.   Static Conformance Requirements   (Normative)

The notation used in this appendix is specified in [CREQ].

A.1. WAP Billing Framework Functions

A.1.1.   Charging Data Providers

Item	Function	Reference	Status	Requirement
WBF-CDR-S-001	*Charging Data Provider*	6.1	M	WBF-PUL-S-001 OR WBF-PUL-S-003 OR WBF-SUB-S-001 OR WBF-SUB-S-002 OR WBF-DEL-S-001 OR WBF-CAN-S-001 OR WBF-QRY-S-001
WBF-CDR-S-002	Tagged Based Header Mechanism	6.1	O	WBF-PUL-S-001 AND WBF-PUL-S-002

A.1.2.   Usage Recording

Item	Function	Reference	Status	Requirement
WBF-UR-S-001	Usage Recording	6.1	M	WBF-CDR-S-001

A.2. Chargeable operations

Item	Function	Reference	Status	Requirement
WBF-PUL-S-001	Content Pull	6.2.2 & 6.2.3.1	O	(WBF-PUL-S-004 OR WBF-PUL-S-005) AND (WBF-PUL-S-006 OR WBF-PUL-S-007)
WBF-PUL-S-002	Recording of Tag-based information	6.2.2 & 6.2.3.2	O	
WBF-PUL-S-003	Content Provider Recording	6.2.2 & 6.2.3.3	O	(WBF-PUL-S-004 OR WBF-PUL-S-005) AND (WBF-PUL-S-006 OR WBF-PUL-S-007)
WBF-PUL-S-004	*Event based single Recording*	6.2.3	O	
WBF-PUL-S-005	*Session based multiple Recording*	6.2.3	O	
WBF-PUL-S-006	Price information provided by *content provider*	6.3.3.2 & 6.2.3.3	O	
WBF-PUL-S-007	Price determined by Charging Control	6.3.3.2 & 6.2.3.3	O	
WBF-SUB-S-001	Push Submission	6.2.2 & 6.2.4.1	O	
WBF-SUB-S-002	Push Replacement	6.2.2 & 6.2.4.1	O	WBF-SUB-S-001
WBF-DEL-S-001	Push Message Delivery	6.2.2 & 6.2.4.2	O	
WBF-CAN-S-001	Push Cancellation	6.2.2 & 6.2.4.3	O	
WBF-QRY-S-001	Push Query	6.2.2 & 6.2.4.4	O	

A.3. CDR format

Item	Function	Reference	Status	Requirement
WBF-CDR-S-001	WAP Record	6.2	M	

Appendix B.   Change History   (Informative)

Type of Change	Date	Section	Description
Class 0	21-November-2002		The initial version of this document.