RCS Presence Best Practice Optimization Guidelines

Version 1.0

25 September 2013

This is a Non-binding Permanent Reference Document of the GSMA

Security Classification: Non-confidential

Copyright Notice

Disclaimer

Antitrust Notice

Table of Contents

1 Introduction

1.1 Overview

This document provides Mobile Network Operators and Vendors with guidelines on best practices, including recommended presence optimizations, which are applicable to OMA SIMPLE PRESENCE [Presence], [Presence2.0_TS] based services as defined in the RCS specifications. These recommendations are provided to support the successful deployment of RCS presence based services per the specific functional and technical service requirements and guidelines as defined in the RCS 5.1 UNI [RCS5.1] and supporting specifications.

1.2 Scope

RCS specifications enable a variety of different solutions and implementations to support differing regulatory, regional, and service provider marketing priorities and requirements. The varying implementations may not necessarily all be compatible with one another without interworking, and may lead to different user experiences. Additionally, the complexity of providing a variety of solutions leads to escalating development and validation costs for successful implementation and deployment.

Consideration is provided in this document for best practices that will help to provide an optimal end-to-end RCS deployment and user experience. Included among the considerations are techniques for traffic congestion management, and consideration of client related issues, two significant issues that need to be considered for successful commercial deployment.

1.3 Definitions

Same definitions from OMA SIMPLE PRESENCE [Presence], [Presence2.0_TS] apply to this document.

1.4 Abbreviations

Term	Description
3GPP	3rd Generation Partnership Project
AP	Aggregation Proxy
API	Application Programming Interface
BA	Broadband Access
CD	Capability Discovery
CNP	Cross-Network Proxy
CU	Capability Update
DDS	Data Definition Specification
EAB	Enhanced Address Book
ETag	Entity Tag
IETF	Internet Engineering Task Force
GSMA	GSM Association

Term	Description
IMS	Internet Protocol Multimedia Subsystem
NAB	Network Address Book
NNI	Network-Network Interface
MSISDN	Mobile Subscriber Integrated Services Digital Network Number
NOT	Notify
OMA	Open Mobile Alliance
OTA	Over-the-air
P2P	Peer to Peer
PC	Personal Computer
PIDF	Presence Information Data Format
PRD	Permanent Reference Document
PUB	Publish
PRS	Presence SIMPLE
PNA	Presence Network Agent
PS	Presence Server
PUA	Presence User Agent
RAN	Radio Access Network
RCS	Rich Communication Suite
RFC	Request For Comments
RLS	Resource List Server
RNC	Radio Network Controller
SIMPLE	Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions
SIP	Session Initiation Protocol
SPI	Social Presence Information
TLS	Transport Layer Security
TS	Technical Specification
UE	User Equipment
UI	User Interface
UNI	User Network Interface
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
US-ASCII	United States - American Standard Code for Information Interchange
UTF-8	UCS Transformation Format—8-bit
VIP	Very Important Person
XCAP	XML Configuration Access Protocol
XDM	XML Document Management
XDMC	XML Document Management Client
XDMS	XML Document Management Server

Term	Description
XML	Extensible Markup Language
XUI	XCAP User Identifier

1.5 References

Ref	Document	Title
[1]	[Presence]	OMA Presence SIMPLE Specification, Approved Version 1.1 – 25 February 2010, http://www.openmobilealliance.org/
[2]	[Presence2.0_DDS]	Presence SIMPLE Data Specification, Approved Version 2.0, 29 September 2009 http://www.openmobilealliance.org/
[3]	[Presence2.1_DDS]	Presence SIMPLE Data Specification, Approved Version 2.1, 02 October 2010 http://www.openmobilealliance.org/
[4]	[Presence2.0_TS]	Presence SIMPLE Specification, Approved Version 2.0 – 10 July 2012 http://www.openmobilealliance.org/
[5]	[Presence2.0_RLS_TS]	Resource List Server (RLS) Specification, Approved Version 1.0 - 10 July 2012, http://www.openmobilealliance.org/
[6]	[PRESENCEIG]	Implementation Guidelines for OMA Presence SIMPLE v1.1 – 09 December 2008, http://www.openmobilealliance.org/
[7]	[PresenceXDM]	Presence XDM Specification, Approved Version 2.0 – 10 July 2012, http://www.openmobilealliance.org/
[8]	[RLSXDM]	Resource List Server (RLS) XDM Specification Approved Version 2.0 – 10 July 2012, http://www.openmobilealliance.org/
[9]	[Presence_Content]	Presence Content XDM Specification, Approved Version 1.0, 10 July 2012, http://www.openmobilealliance.org/
[10]	[SHARED-XDM]	Shared XDM Specification, Approved Version 1.1 – 27 Jun 2008 http://www.openmobilealliance.org/
[11]	[XDM1.1_Core]	XML Document Management (XDM) Specification, Approved Version 1.1, 27 June 2008 http://www.openmobilealliance.org/
[12]	[XDM2.0_Core]	XML Document Management (XDM) Specification, Approved Version 2.0, 03 April 2012, http://www.openmobilealliance.org/
[13]	[RCS5.1]	GSMA RCS 5.1 - Advanced Communications: Services and Client

Ref	Document	Title
		Specification http://www.gsma.com/
[14]	[PRD-IR.90]	GSMA PRD IR.90 - "RCS Interworking Guidelines" http://www.gsma.com/
[15]	[OMA-PRS1.1_Guide]	RCS Presence service Technical Realization (GSMA) – OMA Presence SIMPLE Specification, 1.1; Implementation Guidelines for OMA Presence SIMPLE v1.1 Presence, http://www.openmobilealliance.org/
[16]	[3GPP TS 24.229]	3GPP TS 24.229 Release 10, 3rd Generation Partnership Project; IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP) http://www.3gpp.org

2 Service Overview and General Considerations

2.1 Presence Based Services and the RCS Enhanced Phonebook

In order to allow all RCS users to determine at a glance the list of RCS services that can be used to communicate with their contacts, the Service Capabilities of their RCS contacts are always shown to the other RCS users through interfaces such as the RCS Enhanced Address Book (EAB) illustrated in Figure 1 below.

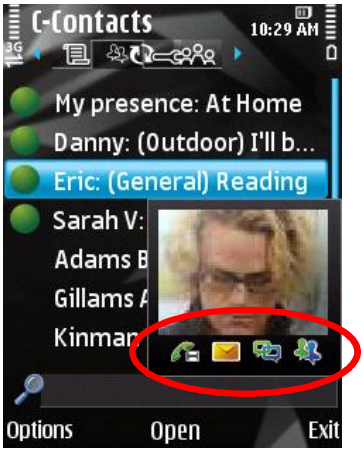

Figure 1: RCS Enhanced Address Book

Service capabilities of the clients that belong to one RCS user are combined at the Presence Server to a single set of service capability information comprising information of all the devices/clients that the user has. As an example, if a user has support for the Video service in his/her PC client but not from his/her mobile, then the service capability information of the user still shows the capability for the Video service.

Sharing an RCS user's service capability information with other RCS users through the device / client EAB as shown above, or in other client and device contexts such as messaging interfaces, is a critical aspect of the user experience. Presence information is

used for Service Capabilities and SPI (Social Presence Information) services which can be distributed in two different ways as defined in the RCS 5.1 UNI specification [RCS5.1]:

- Using “Capability and new user discovery mechanisms” as defined in Section 2.6 of [RCS5.1].
- Using “Social Presence Information” as defined in Section 3.7 of [RCS5.1]

This document provides best practices guidelines to support an operator deployment of these capabilities using an OMA SIMPLE PRESENCE [Presence], [Presence2.0_TS]. Multiple aspects of best practices, to include the optimization of presence based service related traffic are discussed. These include:

- **Traffic reduction**: Methods that can reduce the count of messages
- **Message efficiency**: Methods that can reduce the size of messages (example: compression)
- **Server specific efficiencies**: Techniques that can reduce the processing load / delay at the Presence Server
- **Accuracy / effectiveness**: techniques that can increase presence information accuracy, reduce or eliminate data race conditions

These optimizations are considered from the following perspectives:

- Device and UNI (User Network Interface)
- IMS (Internet Protocol Multimedia Subsystem) Network
- Application Server

2.2 Common Presence based service overview and considerations

2.2.1 Publishing presence information

Both the RCS Capability Discovery/Capability Update (CD/CU) and SPI (Social Presence Information) services require presence sources to publish information into the Presence Server. The discussion below focuses on SIP (Session Initiation Protocol) based publication from a Presence User Agent PUA (Presence Source) (mobile or BA (Broadband Access) UE as defined in [RCS5.1]). The PUA Presence Watcher functionality is captured in sect 2.3 and 2.4.

2.2.1.1 Expires Interval considerations

The value of the Expires header field used for a PUBLISH impacts network traffic in the following ways:

- A shorter expires interval will result in a higher refresh publish traffic rate.
- A longer expires interval will result in a lower refresh publish traffic rate.
- A longer expires interval increases the length of time information is persisted in the Presence Server, and thus the risk that the Presence Server returns stale data in cases where devices are unable to update or unpublish their presence information.

It is not necessary for all endpoints to use the same refresh interval. It may be desirable to have endpoints with a more stable network environment and less volatility in presence

information. In such case, we recommend using a longer refresh interval than for devices with less stable or predictable network connectivity, or a high rate of change in presence information.

The operator must consider many factors to define the value of the Expires header field that best complements their service, network, device, and user experience needs.

2.2.1.2 ETags

The use of ETags as defined in RFC 3903 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1] is a critical aspect of presence service optimization:

- It allows for minimizing the amount of information sent in a publish request.
- In the case of multiple PUA/PNA sources, the use of the ETag establishes the context that allows the PS (Presence Server) to map which updates being received should be applied to which candidate documents.

2.2.1.3 Initial Publish

An initial PUBLISH meets one of the following criteria:

- A publication for the presence event that is formed and sent by the UE, subsequent to device initialization and registration, that contains a full presence document and does not refer to an ETag obtained for a previous publish.
- A publication for the presence event that is formed and sent by the UE, subsequent to device initialization and registration, that does not contain a full presence document and does refer to an ETag obtained for a previous publish.
- A publication for the presence event (that could have been an initial, update, or refresh publish) where an ETag was offered, but the PS response indicated that the ETag was not valid and that an initial publish is required.

2.2.1.4 Update

A PUBLISH Updates occurs when there has been a change to the presence information for a presentity. The change may be caused by:

- Device network status or other changes that require an update to the set of service definitions for that device or endpoint
- Location or other device initiated changes that results in a change to the SPI information
- User initiated changes to their SPI (Social Presence Information) or any other user settable presence information that requires an update.

A PUBLISH update may change a portion of, or all of, the presence information being reported by the device. If the PUBLISH update only changes a portion of the presence information, the operator may choose to implement partial publication.

2.2.1.5 Refresh

A PUBLISH Refresh is the condition where:

- The UE / client has an ETag from a successful previous publish that has not yet expired.

- Presence information has not changed since the last PUBLISH, and the UE wishes to extend the expiry time of their presence information without changing any of that data.

Note: The mechanism to refresh a PUBLISH without sending a body is defined in RFC 3903, endorsed by [Presence], [Presence2.0_TS] and [RCS5.1].

Refresh Interval considerations:

- Different device implementations may provide different rules to use for the point at which a refresh is generated. For example, if a device publishes with a 2 hour expires timer, but uses a 50% rule to trigger a refresh, the effective refresh rate becomes 1 hour.
- An optimal balance will provide sufficient time for the device to successfully refresh without excessively increasing the effective refresh rate or negatively impacting battery life.

2.2.1.6 Un-publish

"Unpublishing" is the use of a PUBLISH request to remove event state information from the presence server.

- This is desirable in the case where a device or endpoint is being powered down, disabled, or is transitioning to a network connectivity state that impacts the presence information stored at the presence server. Most notably service capability information, may inaccurately imply service availability if the information is not removed.
- Service-id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel
- Version: 1.0
- If the information is not "unpublished" as part of client deactivation or de-registration, the stale information (if not "unpublished") would remain at the PS until the expires time is reached for that presence information.

As stated in RFC 3903 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1], an unpublish is effectively "…a publication refresh suggesting an infinitesimal expiration interval. Consequently, the refreshed event state expires immediately after being refreshed."

2.2.2 Service descriptions

The RCS compliant UE /client is expected to fully comply with the [Presence2.0_DDS] schema and definition of <service-description> element and child elements, as appropriate for operator policy and device capabilities and configurations, as specified in Section 2.6.1.2.5.1 Service-descriptions for the Selected RCS Services in [RCS5.1].

2.2.2.1.1 Service definitions from one device

Service definitions originating from the same device SHOULD be aggregated into one tuple when the values are the same in the service-id and respectively the version element.

EXAMPLE: A device that has a GSMA IR.92 voice service and also a GSMA IR.94 video service as endorsed by [RCS5.1], which are both using same

service-id and version but different media capabilities would publish one service tuple i.e. the one for GSMA IR.94:

Service-Id: org.3gpp.urn:urn-7:3gpp-service.ims.icsi.mmtel
Version: 1.0
Servcaps: audio, video, duplex

2.2.3 Multi-Device considerations

The publication of RCS services is done via SIP PUBLISH from each user's device to the Presence Server (PS). Presence composition policies specified in [OMA PRS] define rules that apply to publications coming from various Presence Sources (e.g. RCS clients).

Thus it is expected that any SUBSCRIBE, whether it be for Capability Discovery (CD), Capability Update (CU), or Social Presence Information (SPI), will result in a SIP NOTIFY that contains a presence document that reflects the composed view which integrates presence information from all presence sources, contingent upon the OMA defined composition policy, as well as any operator specific policy.

2.3 User Capability Discovery and Capability Update using Presence (CD/CU)

2.3.1 General Service principles

User Capabilities Discovery (CD) and Capability Update (CU) using Presence are two components of the service enabler "Capability and new user discovery mechanisms" as defined in Section 2.6 of [RCS5.1].

- The "User discovery mechanism" via Presence as defined in section 2.6.2 of the [PRD-IR.90] (aka RCS 5.1 NNI) specification. This mechanism is used in the following contexts:
  - When an RCS device is started for the first time. All contacts in the address book need to be polled
  - Whenever a contact is added to the device address book or whenever a contact's details are edited
  - Optionally, all of the contacts in the address book may be polled using service provider managed parameters (such as POLLING PERIOD)
  - The associated capability information of contacts is older than the configured in RCS parameter CAPABILITY INFO EXPIRY
- Using "Capability update for services" (CU) via Presence as defined in section 2.6.3 of the [PRD-IR.90] specification. This mechanism may be triggered by a number of activities many of which are driven by the local RCS user interacting with specific contacts on their device.

Each of these two components, CD and CU, are discussed separately below.

Each active device and client for an RCS user must publish and maintain their current service capabilities at the presence server. Best Practices regarding publishing and refreshing information is addressed in section 2.2.1, "Publishing Presence Information," above.

2.3.2 User Capability Discovery (CD)

CD is used in several main contexts:

- Where a specific contact is being added or updated in the device EAB
- Where the device or client desires to get updated capability information for all contacts, such as first time device initialization or if the operator has implemented periodic EAB polling;
- Live capability/service discovery for non-VIP contacts when a communication is about to happen.

2.3.2.1 Subscribing to presence information

As defined above, there are different situations where CD is required which differ as follows:

- Service information is only required for one contact. This is the case for adding or modifying a single contact scenario. In this case a single anonymous fetch SUBSCRIBE will be triggered.
- Service information is required for multiple contacts. This is the case for the “first time initialization” scenario, as well as in support of optional periodic EAB polling if supported by the operator. In these cases, following approaches to SUBSCRIBE are available:
    - for the users with a SPI relationship in place, using SPI mechanisms as defined in sect. 3.7.4 of [RCS5.1];
    - For contacts without an SPI relationship:
        - Using a single anonymous fetch SUBSCRIBE for each contact, or;
        - Using an anonymous fetch SUBSCRIBE with request contained list [Presence2.0_TS] as defined in 2.6.2.5 of the RCS 5.1 UNI Specification which states:

        “…the capability query could be initiated by the device using a request contained list that is decomposed by the RLS service in the originating Network.”

2.3.2.1.1 Single anonymous fetch SUBSCRIBE

In those cases where a single anonymous fetch subscribe is triggered, the SUBSCRIBE is expected to have an EXPIRES time of “0.”

2.3.2.2 Anonymous fetch using request contained list SUBSCRIBE

An operator may choose to use a SUBSCRIBE with request contained lists per [Presence2.0_TS]. Considerations:

- This might simplify client logic regarding implementation of periodic polling; rather than having to track when each contact was last fetched the device could simply send list SUBSCRIBES at the appropriate interval to check the whole phone book.
- Contingent upon Operator policy:

  - a single SUBSCRIBE could include up to hundreds of URIs , so it is advisable that maximum list size is configurable;
  - a very short, non-zero expires time may be used when a large number of SUBSCRIBE requests are issued, to allow an additional processing time and multiple NOTIFY responses. If this approach is used, it is desirable that any resulting SUBSCRIBE requests, especially those that will be visible at the NNI, have an expires time of "0" ;
  - Segmentation of very large contacts lists may be achieved by managing the configurable maximum list size. Clients would need to be aware that if the local contacts list exceeds this size, multiple SUBSCRIBE requests are required.

- per RFC 5367 "endorsed by [Presence], [Presence2.0_TS] and [RCS5.1]", and assuming a zero or near zero expires time, coupled with server based throttling and partial list notification as described in Section 3 below, a CD could be performed for a large number of contacts using a single SUBSCRIBE and a relatively small number of Notifies. Example for a case where 100 contacts need to be checked:

  - Using single fetches, checking 100 contacts would require 100 SUBSCRIBE requests and the corresponding responses + NOTIFY requests as appropriate, each of which cross the RAN if the UE is a mobile device
  - A single request contained list fetch could be performed with all 100 URIs in a single subscribe, and it would be expected that only a small number of Notify messages (from 2 – 4) would contain all of the information necessary for all contacts.
  - Thus in this example, the decrease in traffic at the RAN may go from 300 Messages (assuming 50% of the contacts result in a NOTIFY/OK) to 8 – 12 messages.

- The operator deploying this option must also consider:

  - Impact on IMS core
  - Increase in size of messages being sent between UE and PS/RLS.

- Operators that chose to implement this option often use additional optimization strategies, such as Gzip and server based throttling as described in Section 3, to ameliorate the impacts of large SUBSCRIBE messages and many NOTIFY messages.

2.3.3 Capability Update (CU)

A capability update for RCS contact(s) is triggered by one of the following activities:

- After first time registration to obtain the registration state and default set of capabilities for each contact in the device’s address book
- When checking the available RCS services/capabilities to communicate with another user (e.g. from the address book and call-log)
- After establishing voice call to obtain the real-time capabilities for the call, or Chat session provided this has not been performed before, or content sharing during a call is supported.

- After the call returns to an active state (e.g. returning from call wait, call on hold or multiparty call).
- When a communication is active with a user, to provide an update when the relevant available capabilities change.

CU is the simpler of the CD/CU services in that only single fetches are performed based on the client implementation. For contacts without a SPI relationship, refer to section 2.3.2.1.1, “Single anonymous fetch SUBSCRIBE” above.

2.4 Social Presence Information (SPI)

2.4.1 General SPI Service principles

2.4.1.1 Frequency of presence updates

RCS introduced the concepts of VIP contacts and non-VIP contacts, and the difference consists in the frequency of presence information updates (including service capabilities) towards the watchers clients:

- VIP contacts information is obtained through the use of long lived SUBSCRIBE to the VIP contact lists. These SUBSCRIBEs are normally established and maintained (Refreshed) during the period that the RCS EAB at the UE is functioning
- Non-VIP contact information is obtained through the use of fetch SUBSCRIBEs to the non-VIP contact list. While the RCS specification implies that any non-VIP fetch would be performed against the full non-VIP contact lists, it is recommended that client implementations be allowed the flexibility to perform such fetches only for a single contact, or a subset of the contacts, for which information is required with respect to the user’s current needs. Therefore, if information is only required for one contact it may not be desirable to have the UE perform a fetch against the full list of contacts.

Near real-time updates for VIP contacts are process intensive, as they aim to keep the watcher’s clients up-to-date with any changes in the VIP contacts’ presence information regardless if the watcher user is using the RCS client (or browsing the address book) or not.

In the case where the VIP contacts subscription mechanism is employed, the use of optimizations that can reduce traffic should be implemented to the maximum extent practical. For example, these include:

- Conditional Event Notification to suppress redundant presence information in NOTIFY messages when a SUBSCRIBE is refreshed.
- Compression of SIP NOTIFY for list SUBSCRIBEs
- Partial List Notification for list SUBSCRIBES
- Server Based Notification Throttling

If the VIP mechanism is desired, the number of VIP contacts should be limited, as users have typically only a limited number of frequent/favourite contacts. It is recommended that client implementations provide a configuration parameter to control the number of VIP contacts.

There may be an issue for the terminal implementation to determine when to do the pull, but the optimization may be high and the complexity of the function is quite low.

Note that it is possible and sometimes beneficial to use different methods for presence data and watcher info data which is notified with much less frequency.

2.4.1.2 Publishing SPI (Social Presence Information)

In an SPI client:

- SPI information is generally “published” as permanent presence information over the Ut interface;
- Service information is published using SIP PUBLISH;
- A device should not publish SPI information until after the receipt of a NOTIFY in response to a successful subscription to its’ own presence information. This NOTIFY, with the user's own SPI, should arrive during the course of device initialization. This approach will help avoid the following situation:
    - User has devices A and B, and sets note to “Happy” on Device A;
    - Device B is turned off;
    - User changes note to “Sad” on Device A;
    - Device B initializes and before getting a NOTIFY with updated SPI, it sends a publish that changes note from “Sad” to previous value “Happy”

2.4.2 XCAP Optimizations

2.4.2.1 Conditional Operations with E-Tags

- The usage of ETags for XCAP operations is defined by RFC4825 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1].
- The use of ETags with XCAP operations is an important aspect of maintaining document state and allowing for maximum network efficiencies.
    - In the case of multiple XCAP clients accessing the same resource, the XDM server can differentiate invalid requests from valid ones via the entity tag (ETag).
    - ETags also allow minimize network bandwidth if a client fetches a document and the local cache is still valid, the server will respond with a 304 and no body in the request to indicate that the locally cached document by the client is still valid.
- The SPI client should use the “If-Match” header for the PUT and DELETE operations when updating a document. This will avoid conflict if multiple devices are updating the same XDMS list.
- The SPI client should use the “If-None-Match” header for any GET operations when the client fetches a document that the XDMC already has cached locally. If the ETags match, confirming locally cached copy is up to date, then the server will not return a body. This will minimize network bandwidth and depending on implementation, the XDMS may not have to fetch the data from a persistent data store.

2.4.2.2 Compression/Decompression

- The SPI client shall support compression of XCAP PUT and decompression of XCAP GET responses as defined by RFC 2616 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1]. The SPI client should “learn” what type of compression, if any, is supported upon receiving the first successful XCAP GET from the server.

2.4.2.3 Client throttling

- It is recommended that the SPI client support aggregation of XCAP PUT operations. For example, if a user adds one a contact and marks it a VIP, then immediately afterwards adds another contact and marks is as a VIP as well, then the client should aggregate the changes within a given time period to limit the amount of traffic on the network and SIP Notifications that are sent out to the watchers.
- The value of the aggregation interval used for a XCAP PUT requests has impacts on network traffic and server load in the following ways:

    - No aggregation will trigger more notifications to the watchers if multiple contacts are added within a short period (e.g., an import from an external source) or a user that makes frequent tagline updates that are erroneous.
    - A shorter aggregation timer will result in a lower notification rate when multiple contacts are added/ edited and still provide benefits on network traffic and server load.
    - A longer aggregation timer will result in a much lower watcher notification rate, however there is a greater chance of a conflict of updates from multiple sources.

The operator should consider these factors to define an aggregation interval.

2.4.2.4 Encoding of SIP URI’s

- When including sip URI’s within XCAP requests, the SPI client should not percent encode unreserved characters, as defined by RFC3986 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1].
- When percent encoding is required for reserved characters as defined by RFC3986, upper case letters are used for the hexadecimal digits A-F.
- Upper case letters are not used in a SIP URI, with the exception of the ‘userinfo’ part of the SIP URI. Since SIP URI’s are case sensitive this will mitigate future interoperability issues with use of non TEL URI based XUI’s.
- If the SIP URI within an XCAP requests contains characters outside the US-ASCII character set, the SIP URI is first encoded in UTF-8 before being percent-encoded.

2.4.2.5 Encoding of XML Documents

- The SPI client and XDM server should support UTF-8 encoding of documents. UTF-8 is generally regarded as the default character encoding for software applications and it is a superset of all other character sets.

2.4.2.6 Aggregation Proxy and Cross Network Proxy

Aggregation Proxy is defined by [XDM2.0_Core] as the entry point for all XCAP requests from the RCS Client towards XDMSs. The Aggregation Proxy performs the authentication, is

an endpoint for TLS encryption and compression as defined by [XDM2.0_Core] sect. 5.1.1 and sect. 6.3, and aggregates the responses received back from multiple XDMS instances.

In an environment where the Authentication Proxy [24.109] and Aggregation Proxy [XDM2.0_Core] are not realized together, it's up to each operator to implement compression/decompression, TLS terminations and provide UNI interface at the Authentication Proxy or the Aggregation Proxy, as long as the client has one entry point configured for all XCAP requests as per [RCS5.1] via the client configuration parameter XCAP Root URI.

The XCAP NNI interface is provided by the Cross Network Proxy [XDM2.0_Core] as described in [RCS5.1] sect. 3.7.4.8.1.

3 Recommended Optimization Methods for RCS 5.1

This section captures the recommended optimization mechanisms defined in the OMA Presence enabler specification and recommended for RCS presence service, based on methods defined in relevant documents including IETF RFCs and 3GPP specifications. A brief description is given for each recommended optimization method, along with an evaluation of the relevance and applicability of each method.

3.1 Network and device level optimizations

An example of the network and device level considerations that have been used as part of the assessment process for the recommended optimizations is provided in Table 1 below:

Resource / Optimization	Battery	RAN	RNC	Core / IMS	NNI	User experience
Limit number of real time buddies (VIP)	++	++	++	++	++	-
Suspend notifications when inactive	+	-(+)	-(+)	-	0	0
Use Pull instead of Push	+	+	+	0	+	-
Partial Pub / Not	0	+	+	+	0	0
Suppress notification at refresh	+	+	+	+	+	0

Table 1: Network and device level optimizations

Legend: + improvement
- degradation
0 no significant change

3.2 Generic optimizations

3.2.1 Compression (Gzip)

- Description: Use of Gzip compression to reduce size of NOTIFY/XCAP requests (but could be used for other SIP requests as well). Notification encoding is negotiated between server and the client, i.e. if the client supports gzip encoding, Presence Server will gzip all the notifications.
- Standards: RFC 3261 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1] and others
- Impacts:Client, Presence Server, AP (and possibly the XDMS)
- Comments: Efficient compression since PIDF is text based, but dependent on the use case and size of body in the SIP request.

3.2.2 Conditional Operations with E-Tags

- Description: ETag support is defined by RFC4825 endorsed by [XDM2.0_Core] and [RCS5.1] and it is mandatory in both [XDM2.0_Core] and [RCS5.1]. ETags are essential to XML document version tracking, optimization of the Ut traffic and network efficiencies. It is recommended to:

  - use “If-Match” header by client for the XCAP PUT and DELETE.
  - use “If-None-Match” header by client for any XCAP GET operations when the client fetches a document that the XDMC already has cached locally.
  - Use ETag in SIP PUBLISH as per RFC 3903 endorsed by [Presence], [Presence2.0_TS] and [RCS5.1].

- Standards: [Presence], [Presence2.0_TS] (specifically by RFC4825 and RFC 3903 [Presence], [Presence2.0_TS] (specifically by; [XDM2.0_Core]; [RCS5.1]
- Impacts:Client, Presence Server, AP, XDMS
- Comments: needed in multi-device scenarios, minimizes network bandwidth

3.3 Signalling Optimizations

3.3.1 Partial publication (diff)

- Description: Partial publications offer relief on the UNI traffic load as they carry only those parts of presence information that have changed since last publication, rather than the full presence state
- Standards: [Presence2.0_TS] (specifically by RFC 5264 and RFC 5262 endorsed by [Presence2.0_TS] and [RCS5.1])
- Impacts: Client and Presence Server
- Comments: Since device publication for one endpoint is based on RCS presence document which are not large, the expected gains would be marginal.

3.3.2 Partial List Notification

- Description: For list based subscriptions, when a NOTIFY arrives, it can contain full or partial state information. This capability eliminates the requirement for each NOTIFY to provide all presence information for each resource in the list. The partial state NOTIFY may only contain information for those resources for which there has been a

change. This capability eliminates the transmission of redundant presence information.

- Standards: [Presence2.0_TS] (specifically by RFC 4662 endorsed by [Presence], [Presence2.0_TS])
- Impacts: Client and Presence Server
- Comments: relevant to SPI and RFC 5367 list subscriptions endorsed by [Presence], [Presence2.0_TS].

3.3.3 Conditional Event Notification

- Description: This supports the notion of ETag on subscription/notification. This has several benefits:
  - No notification is triggered to the client when it refreshes its subscription and no updates have occurred.
  - No notification between networks is triggered if no updates happen during the refreshing period.
- Standards: [Presence2.0_TS] (specifically by RFC 5839 endorsed by [Presence], [Presence2.0_TS]), [RCS5.1]
- Impacts:Client, RLS and Presence Server
- Comments: Relevant to SPI, more impact when refresh intervals are shorter.

3.3.4 Publication Throttling

- Description: The service provider may configure a Presence Source with the shortest allowed time period between two PUBLISH requests. This can be done with OTA (Over-the-air) Provisioning or local configuration
- Standards:[Presence2.0_TS] sect. 5.1.2.3
- Impacts: Client
- Comments: both the server could do the throttling, and the client can configure it - this can lead to conflicts and is confusing. It is dependent on the client implementation.

3.3.5 Request Contained List Subscription

- Description: The request contained list subscription is used in RCS for Anonymous capability polling for multiple contacts in one shot, and it is a way to create a list with a set of resources and subscribe to it using a single SIP request. This is achieved by including the list of resources (as defined in [RFC5363]) in the body of the SUBSCRIBE request.
- Standards: [Presence2.0_TS], [RCS 5.1] (specifically by RFC 5367 endorsed by [Presence], [Presence2.0_TS])
- Impacts: Client and Presence Server.
- Comments: May have some relevance to CD/CU contingent on how list subscriptions are implemented. Impacts at back-end to be assessed for large lists.

3.3.6 Server based Notification Throttling

- Description: The PS MAY have a local throttling configuration setting that limit the rate at which notifications are generated (i.e. the shortest time period between two NOTIFY requests for a given Watcher).

- Standards: [Presence2.0_TS] sect. 5.5.3.6
- Impacts: Presence Server
- Comments: Relevant for RCS presence overall (CD/CU and SPI).

3.3.7 Limit number of real time buddies (VIP contacts)

- Description: The Client SHOULD have a local configuration parameter to allow the operator to limit the number of VIP contacts for which real-time subscriptions dialogs are maintained.
- Standards: [RCS5.1], [Presence2.0_TS]
- Impacts: Client
- Comments: Relevant for RCS presence overall (CD/CU and SPI).

3.4 XCAP Optimizations

3.4.1 Client throttling

- Description: The user often makes several subsequent updates to a XML document within a short time span. It is recommended that the client aggregates several changes into one XCAP PUT operation of the entire document (e.g. user adding several contacts and marking them VIP). This optimizes both the XCAP traffic as well as the consequent SIP Subscriptions that are sent out to the watchers.
- Standards: [RCS5.1], [XDM2.0_Core] (specifically by RFC4825 endorsed by [XDM2.0_Core]).
- Impacts: Client
- Comments: The value of the throttling timer in the client needs to be carefully set as per network specifics (traffic profile, user patterns, etc).

3.4.2 Encoding of SIP URI's

- Description: Percent encoding of unreserved characters, is defined by RFC3986 endorsed by [XDM2.0_Core]. It is recommended that:
  - the RCS client does not percent encode unreserved characters when including sip URI's within XCAP requests.
  - when percent encoding is required for reserved characters as defined by [RFC3986], the upper case letters are used for the hexadecimal digits A-F.
  - Upper case letters are not used in a SIP URI, with the exception of the 'userinfo' part of the SIP URI. Since SIP URI's are case sensitive, this will mitigate future interoperability issues with use of non-MSISDN based XUI's.
  - If the SIP URI within XCAP requests contains characters outside the US-ASCII character set, the SIP URI is first encoded in UTF-8 before being percent-encoded.
- Standards: [XDM2.0_Core] (specifically by RFC3986 endorsed by [XDM2.0_Core])
- Impacts: Client, XDMS, AP, CNP
- Comments: -

3.4.3 Encoding of XML Documents

- Description: Percent encoding of unreserved characters, is defined by [RFC3986]. It is recommended that:
    - The client and XDM server support UTF-8 encoding of documents. UTF-8 is generally regarded as the default character encoding for software applications and it is a superset of all other character sets
- Standards: [XDM2.0_Core] (specifically by RFC3986 endorsed by [XDM2.0_Core])
- Impacts: Client, Presence XDMSs, AP, CNP
- Comments: -

3.4.4 Aggregation Proxy

- Description: Aggregation Proxy (AP) is defined by [XDM2.0_Core]. The RCS Client is configured with the address of the Aggregation Proxy (XCAP Root URI) towards which all XCAP requests towards XDMSs are addressed. The Aggregation Proxy Client performs the authentication, is an endpoint for TLS encryption and compression as defined by [XDM2.0_Core] sect. 5.1.1 and sect. 6.3, and aggregates the responses received back from multiple XDMS instances, abstracting the underlying network layout of the XDMSs from the clients.
- Standards: [RCS5.1], [XDM2.0_Core].
- Impacts: Client, XDMS.
- Comments: it is possible that the Aggregation Proxy functions are implemented in several boxes (e.g. some functions are delegated into a generic Authentication proxy box). In all cases, the client must still be configured with only one address (XCAP Root URI) for the Aggregation Proxy.

3.4.5 Cross Network Proxy

- Description: Cross Network Proxy is defined by [XDM2.0_Core] and its functions are described in [RCS5.1] sect. 3.7.4.8.1. Cross Network Proxy applies the XCAP NNI SLA policies.
- Standards: [RCS5.1], [XDM2.0_Core].
- Impacts: AP.
- Comments: As per [RCS5.1], all functions of Cross Network Proxy are endorsed except for search related requests.

4 Other Available optimization mechanisms

Other methods and mechanisms provided by Presence related specifications (OMA, 3GPP, IETF) have been assessed and concluded to provide less efficient impacts, such as:

4.1 Partial Notification (diff)

- Description: Partial notifications offer relief on the UNI and NNI traffic load as they carry only those parts of presence information that have changed since last notify, rather than the full presence state.
- Standards: [Presence], [Presence2.0_TS] (specifically by RFC 5263 and RFC 52620
- Impacts: Client and Presence Server

- Comments: Complex to implement in devices, and small efficiency gained when implemented in support of large list based subscriptions.

4.2 Event Notification Cancellation

- Description: The client can ask for transient notification cancellation (e.g. when going to idle mode). The subscription is maintained, and when the client is refreshing its subscription (e.g. when going back to active mode), ETag value is used to check whether updates need to be sent to the client.
- Standards: [Presence], [Presence2.0_TS] (specifically by RFC 5839 endorsed by [Presence], [Presence2.0_TS])
- Impacts: Client and Presence Server
- Comments: Not relevant to CD/CU. Loss and gain case, depending on how phone is used and client implementation.

4.3 Event Notification Filtering

- Description: This RFC describes the operations a subscriber performs in order to put filtering rules associated with a subscription to event notification information in place. Filtering is a mechanism for controlling the content of event notifications. Transportation of the filter to the server is achieved by inserting the XML document, as defined in RFC 4661, in the body of the SUBSCRIBE request. Possible Use Cases:

  - Interested only in location information
  - Interested in getting information about the communication means and contact addresses on which the presentity is currently available for communication.
  - Request triggers to receive notifications based on specific attribute transition

- Standards: [Presence], [Presence2.0_TS] (specifically by RFC 4660/4661 endorsed by [Presence], [Presence2.0_TS])
- Impacts: Client and Presence Server
- Comments: Relevant for more sophisticated Use cases, including APIs and 3rd Party Applications, rather than standard RCS Use Cases.

4.4 Event Notification Throttling / Notification rate control

- Description: This mechanism is subscription based; it means that the rate is enforced (and possibly negotiated) at each subscription.
- Standards: OMA Presence TS sect. 5.1.2.3 (references an obsolete draft, now incorporated into RFC 6446, but the RFC includes deltas from the draft)
- Impacts: Client and Presence Server
- Comments: Variable on Use Case. Difficult to predict the traffic based on each UE value proposal.

4.5 Content Indirection

- Description: Content indirection enables a client to retrieve by HTTP a presence document instead of receiving it in the NOTIFY request. This mechanism allows large documents to bypass Network Elements in the SIP plane, such as an SBC, that may have some limit in the maximum size of the SIP message they can process.

- Standards: [Presence], [Presence2.0_TS] (specifically by RFC 4483)
- Impacts: Client, Presence Server and media repository
- Comments: Variable on Architecture and Use Case. In RCS Presence context, the documents are not so big, hence it may be more consuming to use indirection and have another request to fetch the link, rather than not use indirection.

5 Future Considerations

Opportunity may exist for substantially improved rich presence based signalling with additional refinement of existing methods, and/or the implementation of new methods. Additional consideration is needed, for example, with respect to the following possibilities:

5.1 "Smart" Signalling Aggregation

Multiple presence related signalling messages destined for a given user may be aggregated in the presence server, reducing the number of messages transmitted. "Smart" signalling aggregation considerations can include the importance of the message, delay considerations, variable duration of the aggregation window time, network conditions (e.g. traffic congestion), historic 'burstiness' of communications for the user at a given day of week, time of day, etc.

5.2 User Behaviour based Signalling Optimization

The transmission of presence related signalling messages can take into consideration knowledge of the user's behaviour, and social relationships such that data is not signalled if the data is considered to be less critical, or the user's relationship is less strong or the user isn't active on a given device. As an estimate, this could reduce presence signalling by as much as 94% without significantly impacting the user experience.

5.3 P2P Signalling Manager

Core Network based signalling optimization methods may also be applied to peer to peer signalling between devices if the signalling is intercepted in the core network. Thus additional signalling optimization could be applied for P2P signalling approaches.

5.4 Client Optimizations

There may be additional opportunities for the realization of signalling optimization in device clients, including for example:

- Data Compression, Event Filtering, Partial Information Transmission, Rate Throttling
- Smart Polling based on strength of user's friends' relationships and usage behaviour patterns (e.g. do not query/exchange presence information as frequently for friends without strong relationship ties)
- Smart signalling aggregation of outbound presence information messages

5.5 NAB / PS Interworking

A service provider may deploy a Network Address Book (NAB) that synchronizes data between a user's multiple devices. A user with RCS presence server based services may also have (i.e. social) data synchronized between their multiple devices using the capabilities of the presence server.

It is for further consideration to determine if signalling optimization could effectively be provided by eliminating possibly redundant signalling in multi-device scenarios where data synchronization is provided with both a Network Address Book and an RCS Presence Server.

Document Management

Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
0.1	July 12th 2013	Initial version	RCCTF #69	GSMA
1.0	Aug 9th 2013	Submitted to DAG#108 and PSMC#116 for approval	RCS & PSMC	GSMA

Other Information

Type	Description
Document Owner	RCS
Editor / Company	Henry Bowes, GSMA

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.