The Continue Header Field for the Session Initiation Protocol (SIP)  
draft-sinha-dispatch-sip-continuation-option-01.txt

Abstract

Before placing a call, it is quite often useful for the Caller to  
know whether a Callee is in favourable state to receive call or not.  
This document defines an optional tag "continue" and a header  
"Continue" to address the purpose. The "Continue" header field is  
to confirm the session continuity with the Callee from the Caller  
after an option for session continuity is placed by the Callee based  
on the unfavorable state of the Callee. This functionality is needed  
to resolve the unwillingness of the Callee to receive any call. An  
option is given to the Callee by the Service Provider or by the  
Handset Manufacturer or by the Carrier to establish this requirement.

Status of this Memo

This Internet-Draft is submitted in full conformance with the  
provisions of BCP 78 and BCP 79. Internet-Drafts are working  
documents of the Internet Engineering Task Force (IETF). Note that  
other groups may also distribute working documents as  
Internet-Drafts. The list of current Internet-Drafts is at  

Internet-Drafts are draft documents valid for a maximum of six months  
and may be updated, replaced, or obsoleted by other documents at  
any time. It is inappropriate to use Internet-Drafts as reference  
material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 10, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the  
document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal  
Provisions Relating to IETF Documents  
(http://trustee.ietf.org/license-info) in effect on the date of  
publishation of this document. Please review these documents  
carefully, as they describe your rights and restrictions with  
respect to this document.
1. Introduction

Session Initiation Protocol (SIP) allows Caller to establish sessions with Callee without knowing whether Callee is in a position to accept session request or not. There is no means by which Caller can know the favourable state of Callee. There are several reasons why the Caller wants to know the favourable state of Callee before finally placing the call, because Callee may be in a different time zone, Callee may be in a meeting, theatre, having lunch with family/friends, etc. Callee may not want to be disturbed but may like to receive calls which is urgent and of good interest to him. Another example Bob may not like office calls on weekends or at his personal/family time, but a project which requires his consent to progress or business gets affected can be treated as urgent and he would like to hear them. Same in case he’s in a business meeting and doesn’t like receiving call from family or his team but anything which requires his urgent attention, it would be important for him to accept. The nearest approach or solutions to such problem available till date is described below along with the appropriate cause why this document is not considering this for implementation. These approaches are not good enough to address all concerns and address the purpose of this proposal, which is due to lack of messaging and current implementation capability. Also it doesn’t give Caller to control call based on its urgency of call to establish session with the Callee, provided prior info about Callee in unfavorable state to receive calls, wherein Bob is willing to accept urgent calls.
o By OPTION Method - It is used to determine the capabilities of SIP User Agent (UA). This OPTION method allows a SIP User Agent Client (UAC) to discover information about the supported method, content types, extensions, codec, etc for a client which is registered with the Server. OPTION method is given by the Caller to know either Server Capabilities or Client Capabilities and not its state.

o By Priority Header - Priority Header can be used to override an ongoing call, DND option or any voice feature based on the Priority of the call as set by the Caller and as agreed with the Operator. If Called Party supports the feature and Caller does not support Priority, then this feature will be just DND. The purpose of this document is not to have DND by any means.

o By Do Not Disturb (DND) - When user enable DND on the phone, this parameter allows user to specify how the DND features handle incoming calls:

- Call Reject - This option specifies that no incoming call information gets presented to the user. Depending on how user configure the DND Incoming Call Alert parameter, the phone may play a beep or display a flash notification of the call.

- Ringer Off - This option turns off the ringer, but incoming call information gets presented to the device, so that the user can accept the call.

This feature doesn’t serve requirement of Caller in either of the cases Callee may not notice the alerts and call will be diverted to voicemail. The purpose of this document is not to have DND.

o By Presence Feature - The use of PRESENCE with UAC clients restricted to higher end-mobile devices. It is also not necessary that caller and callee both are in each-other’s buddy’s list or connected to same social-networks as it is not necessary that call is between two known persons.

Therefore this document describes a mechanism for a Callee convey an information to a Caller to place a call only when call is urgent by introducing "Call Continuation Option" and "Non-Dedicated Dot Not Disturb (NDDND)" as two a new features in TELECOMMUNICATION.

Definition

o Call Continuation Option - is a feature where an option is given to the Caller after Caller dials the Callee number to confirm the application whether to finally place the call or not based on certain configuration at the Callee UE or at the Voice Feature Service of Callee.
Non-Dedicated Do Not Disturb (NDDND) - is a state of Callee which portrays that the Callee is in a non-dedicated or slight DND mode with willingness to receive calls only if it is urgent. However the decision is driven by the Caller whether to finally place a call or not. By setting this option at the Callee UE or at voice services of Callee on the operator side, it ensures that the Caller gets a notification after placing its call that Callee is in an unfavorable state to receive call at this moment but in a position to accept any call if caller thinks its urgent.

1.1 Terminology

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 RFC 2119. 

2. Overall Operation

Callee is willing to receive only urgent call by enabling NDDND. This will convey information to Caller that Callee is not in a favorable state to receive calls and option will be given to Caller to place a call only if it is urgent. This is achieved adding an optional tag "continue" in the Require header of 182 Queued provisional response from Callee. This response from Callee will be generated only if the Caller has the capability to support this functionality otherwise this feature at Callee would be suppressed and 180 Ringing will be sent as response on receiving INVITE.

Caller is keen to know whether the Callee is free or not to take a call. So Caller sends INVITE with option tag continue in the Supported. The Callee interprets the Caller up-to-date and inline capabilities and responds with 182 Queued SIP response with header Require: continue if the feature NDDND is set at the Callee. This SIP response is interpreted at the Caller with a message popup or some in call information (device implementation) telling the unfavorable state of Callee to receive calls but tells its availability state also with an "Y" or "N" which can also be simulated by Hard Call Place GREEN Button or Call Cancel RED Button as per Caller wishes to perform the needful action. This is described in details in the mentioned use cases in Section 7. Accordingly PRACK [3] with Continue header as Yes (if 'Y' option is selected or GREEN button is pressed) or No (if 'N' option is selected) to be sent to Callee or the CANCEL request from Caller will be initiated if Hard RED Button is pressed.

The Caller will have following 3 options to the control session.

(i) Continue: Yes

   Caller MUST inject Continue header with field value as "yes" or "YES" in the PRACK to support the Require header of provisional response.
(ii) Continue: No

Caller MUST inject Continue header with field value as "no" or "NO" in the PRACK to support the Require header of provisional response.

(iii) Caller Hungs Up

CANCEL is sent by Caller and normal call flows as per RFC 3261[1] follows based on implementation (based on terminating leg as SIP AS or Called UA) to terminate and clear this call leg.

If the Caller does not act on received 182 Queued provisional responses, retransmission timer implemented as per RFC 3261[1].

---

Figure 1: Algorithm for Call Continuation Option feature

3. UAS Behavior

A UAS MAY send 182 Queued provisional (by adding Require header field with the option tag continue) response if the initial INVITE request
contained a Supported header field with the option tag continue. If the UAS is unwilling to do so, it MUST ignore the option tag continue and proceed as per guidelines described in RFC 3261[1].

A UAS MUST send 182 Queued provisional (by adding Require header field with the option tag continue) response if the initial INVITE request contained a Require header field with the option tag continue. If the UAS is unwilling to do so, UAS responds with 180 Ringing.

A UAS MUST NOT send 182 Queued provisional (by adding Require header field with the option tag continue) response if the initial INVITE request did not include either a Supported or Require header field indicating this feature.

A UAS MUST NOT send 182 Queued provisional (by adding Require header field with the option tag continue) response if the initial INVITE request include Priority header with high value.

If more than one Continue header field is present in PRACK with two different field values, the UAS MUST reject the request with a 400 Bad Request response.

If a Continue header field is present in a request other than PRACK, the UAS MUST reject the request with a 400 Bad Request response.

4. UAC Behavior

When the UAC creates a new request, it can insist for Call Continuation Option in the provisional response for the request. To do that, it inserts a Require header with the value continue as option tag in the request. A Require header with the value continue MUST NOT be present in any requests except INVITE.

If the UAC does not wish to insist on usage of Call Continuation Option in the provisional response, a Supported header MUST be include in the request with the option tag continue. The UAC SHOULD include this in all INVITE requests.

If a provisional response is received for an initial request and that response contains a Require header field containing the option tag continue, the response is send in field value of Continue header in PRACK message.

If a provisional response is received for an initial request and that response contains a Require header field containing the option tag continue and UAC is unwilling to establish a session, it MUST reject with CANCEL message.
5. The Continue Header Definition

The Call Continuation Option feature makes use of Continue header which provide control to Caller on establishing the session.

The Continue header field MAY appear as an option extension header in PRACK request of the Call Continuation Option feature. The syntax of Continue header field follows the standard SIP parameter syntax.

```
Continue = "Continue" HCOLON continue_value
continue_value = "YES" / "yes" / "NO" / "no"
```

So ideally optionally used "Continue" header in SIP PRACK request would look like

```
Continue: "YES"
Continue: "yes"
Continue: "NO"
Continue: "no"
```

The information about Continue header field in this document in relation to method and proxy is summarized in Table 1.

<table>
<thead>
<tr>
<th>Header field where proxy</th>
<th>ACK</th>
<th>BYE</th>
<th>CAN</th>
<th>INV</th>
<th>OPT</th>
<th>REG</th>
<th>PRACK</th>
<th>REFER</th>
<th>SUB</th>
<th>NOT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Header field</td>
<td>R     -    -    -   -   -   -   -    o     -    -   o</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Header field where</th>
<th>INF</th>
<th>UPD</th>
<th>MSG</th>
<th>PUB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continue</td>
<td>R     -    -    -   -   -   -   -   -   -</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Additional Table Entries for the Continue Header

The "where" column describes the request and response types in which the header field may be used. The header may not appear in other types of SIP messages. Values in the where column is:

- R: header field may only appear in requests.
- m: the header field is mandatory.
- o: the header field is optional.
- -: the header field is not applicable.
6. Backwards Compatibility

This feature or SIP implementation does not have any impact on existing Network designs and Handsets. This draft proposes the use of additional header call SIP "Continue" header in order to incorporate this feature of NDDND.

Handsets or Network who do not understand this feature shall not handle and normal SIP behavior follows as per RFC 3261.

There are 3 use cases for backward compatibility:

- Caller Not Updated, Callee Updated
- Caller Updated, Callee Not Updated
- Both Caller and Callee Not Updated

Handling of SIP messages for the above mentioned scenarios are clearly mentioned in Section 3.

7. Examples and Use cases

This section contains a number of examples and use cases that illustrate the use of the Continue header field. For simplicity, header fields are usually shown in the same order. Usually only the minimum required header field set is shown. Also, message body content lengths are often not calculated, but instead shown as "..." where the actual octet count would be.

Messages are identified in the figures as F1, F2, F3, etc. This references the message details in the table that follows the figure.

Comments in the message details are shown in the following form:
/* Comments. */

7.1 Call is Finally Placed by Calling Party

In this scenario, Alice has enable NDDND feature in her profile. The Bob sends initial INVITE with option tag continue in Support header. Alice asked for confirmation of urgent call before ringing by 182 Queued provisional response with option tag continue in Require header. Bob confirmed as urgent call by providing yes as Continue header field value. Alice’s phone rings finally.
### Message Details

/* The initial INVITE request has option tag continue in Support header */

**F1 Message Bob->Proxy**

```
INVITE sip:alice@proxy.com SIP/2.0
Via: SIP/2.0/UDP client.biloxi.example.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Alice <sip:alice@atlanta.example.com>
From: Bob <sip:bob@biloxi.example.com>;tag=67890
Call-ID: a84b4c76e66710
CSeq: 1 INVITE
Supported:100rel,continue
Contact: <sip:bob@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: ...
```

(Bob’s SDP not shown)

---

**Figure 2: Call is Finally Placed by Calling Party**

Bob

<table>
<thead>
<tr>
<th>INVITE   F1</th>
<th>Proxy      </th>
<th>Alice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100 Trying   F2  </td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-----------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-----------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>182 Queued   F5  </td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-----------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PRACK   F6    </td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>200 OK   F9    </td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-----------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>180 Ringing   F11  </td>
<td></td>
</tr>
</tbody>
</table>

* Rest of flow not shown *

---

* De_Sinha

Expires March 2012
/* Alice supporting the new header extension=continue, initially her phone does not ring upon receiving initial INVITE request. It will include option tag continue in Require header while sending 182 Queued provisional response to Bob. The response, F5, received at Bob, might look like this*/

F5 Message Proxy ->Bob

SIP/2.0 182 Queued
Via: SIP/2.0/UDP client.biloxi.example.com;branch=z9hG4bKnashds8
To: Alice <sip:alice@atlanta.example.com>;tag=12345
From: Bob <sip:bob@biloxi.example.com>;tag=67890
Contact: <sip:bob@192.0.2.4>
Require: 100rel,continue
Call-ID: a84b4c76e66710
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: ...

(Alice’s SDP not shown)

/* Thus Bob comes to know that Alice is not willing to accept call if the call is not very important. So Bob has control on the call and can take a decision on his call. Assume Bob finally place the call include "yes" or "YES" as the field value of Continue header in PRACK method (F6).*/

F6 Message Bob -> Proxy

PRACK sip: sip:alice@proxy.com SIP/2.0
Via: SIP/2.0/UDP client.biloxi.example.com:5060;
branch=z9hG4bKnash009
Max-Forward: 70
From: Bob <sip:bob@biloxi.example.com>;tag=67890
To: Alice <sip:alice@atlanta.example.com>;tag=12345
Call-ID: a84b4c76e66710
Require: 100rel,continue
CSeq: 2 PRACK
RAck: 1 1 INVITE
Continue: YES
Content-Length: 0

Alice’s phone rings as it got confirmation. The guidelines of rest of messages flow is described in RFC 3261.

7.2 Call is Finally Not Placed by Calling Party

In this scenario, Bob finally decided not placed the call to Alice as
his call is not very urgent and Alice is willing to receive only urgent call at this movement. The Alice’s phone does not ring and call is forward to preset destination if any.

<table>
<thead>
<tr>
<th>Bob</th>
<th>Proxy</th>
<th>Alice</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVITE F1</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>100 Trying F2</td>
<td></td>
<td>INVITE F3</td>
</tr>
<tr>
<td>&lt;--------------</td>
<td></td>
<td>182 Queued F4</td>
</tr>
<tr>
<td>182 Queued F5</td>
<td></td>
<td>&lt;--------------</td>
</tr>
<tr>
<td>PRACK F6</td>
<td></td>
<td>PRACK F7</td>
</tr>
<tr>
<td>---------------</td>
<td></td>
<td>200 OK F8</td>
</tr>
<tr>
<td>200 OK F9</td>
<td></td>
<td>486 Busy Here F10</td>
</tr>
<tr>
<td>&lt;--------------</td>
<td></td>
<td>&lt;--------------</td>
</tr>
</tbody>
</table>

* Rest of flow not shown *

Figure 3: Call is Finally Not Placed by Calling Party

/* Assume Bob finally wish not to place the call with Alice as his call is not very important. So he injects "no" or "NO" as the field value of Continue header in PRACK method (F6)*/

F6 Message Bob -> Proxy

PRACK sip: sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP client.biloxi.example.com:5060;branch=z9hG4bK124
Max-Forward: 70
From: Bob sip:bob@biloxi.example.com;tag=67890
To: Alice <sip:alice@atlanta.example.com>;tag=12345
Call-ID: 12ka4@biloxi.example.com
CSeq: 2 PRACK
Continue: NO
Content-Length: 0

Alice’s phone does not ring as it got confirmation as no. PRACK is responded by 200 OK followed by 486 Busy Here as per RFC 3261.
7.3 Caller Hungs Up

In this example, Bob hangs up the call with Alice when he comes to know that Alice need confirmation about importance of call before ringing, which is shown in Figure 3.

```
Bob    | Proxy    | Alice
INVITE | F1       |
       |<---------------|---------------------->
100 Trying | F2       |
       |----------------------| INVITE | F3       |
       |<----------------------| 182 Queued | F4     |
       |----------------------|<----------------------| 182 Queued | F5     |
       |----------------------|----------------------| CANCEL | F6     |
       |<----------------------|----------------------| CANCEL | F7     |
* Rest of flow not shown *
```

Figure 4: Caller Hungs Up

/* Bob disconnects by initiating a Cancel (F6) request and the call is handled as per RFC 3261[1]*/

```
F6 Message Bob -> Proxy
CANCEL sip: sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK123
Max-Forward: 70
From: Bob <sip:bob@biloxi.example.com>;tag=67890
To: Alice <sip:alice@atlanta.example.com>;tag=12345
Call-ID: 12ka4@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0
```

The guidelines of rest of messages flow is described in RFC 3261[1].

7.4 Caller does not has Call Continuation Option feature

There might be cases when Bob doesn’t support Continue header but Alice has enable NDDND feature. In such a case Bob wouldn’t be facilitated with continue advantages. Bob sends initial INVITE without option tag continue in Support or Require header and will receive 480 (Temporarily Unavailable) responses. Alice phone will not ring.
NDDND feature will be overwritten and Alice phone ring if the initial INVITE request include Priority header with high value. Basic intention of Alice is not to block urgent or emergency call but to avoid unimportant call as busy.

7.5 Callee does not support NDDND feature

Alice phone rings which even though option tag continue present in Support header of initial INVITE send by Bob.

8. Implementation Recommendation

"Session Continuation Option" feature in SIP is a new call control feature proposed in this document applicable in favour of Callee that gives option to Caller requesting for a confirmation to place a call to Callee based on the following new applications at the Callee.

- Callee Unfavourable Time to Receive Call (usually when the
Callee is sleeping or travelling, based on Callee time zone which is based on his location that needs to be updated by Callee in case of VoIP subscriber, and manually or automatically done by Mobile Service Provider based on Daylight Savings for 3G/4G subscriber).

- Callee has set a Non-Dedicated Do-Not-Disturb (NDDND) that is DND with lower priority when the Callee is in a theatre, meeting, driving, lunch/dinner, etc. - Callee is willing to receive the call only when it is very urgent.

This application is totally based on SIP Call Session Establishment principles (Offer-Answer Model) with minor deviation in call flows and is implementation specific and driven based on configuration at.

- SIP UA or Voice Over Internet Protocol (VoIP) End Device, 3G/4G Handsets (Device Driven) - Device plays more important role in Call Handling.

- SIP Back-To-Back User Agent (B2BUA) server as a part of Voice Feature Services given by the Service Provider (Service Provider Driven) - Network plays more important role in Call Handling instead of SIP Terminals or 3G/4G Handsets.

This Call Feature Application handling behaviour and subsequent actions at the Calling and Callee and SIP AS is clearly defined later using State Diagrams based on the call confirmation options handled by Calling Party and relevant configurations / services provided by the Network Service Provider.

This feature can be implemented using SIP with minor changes at protocol level by introducing "Continue" header in SIP PRACK Request and slight changes in the basic Offer-Answer Model implementation in both UAs and SIP AS based on implementation, calls of which are discussed in detail later on. There will be an additional need of changes in UA Application and also at SIP AS in case this feature is driven by a SIP AS instead of Callee.

This feature is fully backward compatible and doesn’t have any impact on existing Subscribers, Devices, Network Setup and Operation. The Service Provider may need to upgrade their SIP Application server in case they wish to give this new voice feature as a new service to their subscribers. Similarly Device Manufacturer needs to upgrade the application of their handsets or introduce as a new application feature in their new Handsets or Devices.

9. IANA Considerations
This document registers a new header field name with a compact form and one new option tag.

9.1 IANA Registration of "Continue" SIP Header field

Name of Header: Continue
Compact Form: g

9.2 IANA Registration of "continue" SIP Option-tag

Name of option: continue
Description: Support for the SIP Continue header

10. Security Considerations

The Continue header in this document does not in itself have security considerations. However, as mentioned in RFC 3427, an important reason for the IETF to manage the extensions of SIP is to ensure that all extensions and parameters are able to provide secure usage.

11. Acknowledgements

Thanks to Paul Kyzivat, Worley, Dale R (Dale), John Elwell and Ram Mohan R who provided helpful comments, feedback and suggestions.

Also the authors would like to thank Mayur Saxena, Jay Prakash Dubey, Lakshmi P Vijay Badithe, Prasad Kulkarni, Rajesh Batagurki, Vineet Hada, Ayan Ghosh, Reetesh Gupta, Nishesh Shukla and Dipankar Goswami for their input.

12. Normative References


13. Authors’ Addresses
Amardeep Sinha  
FF02, First Floor,  
Rainbow Residency,  
Green-Glen-Layout,  
Bellandur, Outer-Ring-Road  
Bangalore (Karnataka)  
India -560103  
EMail: amardeep_sinha@rediffmail.com

Subhrajyoti De  
F.E. 91 SECTOR 3  
SALT LAKE CITY  
KOLKATA - 700 106.  
WEST BENGAL. INDIA.  
EMail: de_subhrajyoti@yahoo.co.uk

Sunil Kumar Sinha  
FF01, First Floor,  
Rainbow Residency,  
Green-Glen-Layout,  
Bellandur, Outer-Ring-Road  
Bangalore (Karnataka)  
India -560103  
EMail: sunilkumarsinha9@rediffmail.com