



AT&T Service Developer Guide

In-App-Messaging API: Index Management

Publication Date: February 2015





Legal Disclaimer

This document and the information contained herein (collectively, the "**Information**") is provided to you (both the individual receiving this document and any legal entity on behalf of which such individual is acting) ("**You**" and "**Your**") by AT&T, on behalf of itself and its affiliates ("**AT&T**") for informational purposes only. AT&T is providing the Information to You because AT&T believes the Information may be useful to You. The Information is provided to You solely on the basis that You will be responsible for making Your own assessments of the Information and are advised to verify all representations, statements and information before using or relying upon any of the Information. Although AT&T has exercised reasonable care in providing the Information to You, AT&T does not warrant the accuracy of the Information and is not responsible for any damages arising from Your use of or reliance upon the Information. You further understand and agree that AT&T in no way represents, and You in no way rely on a belief, that AT&T is providing the Information in accordance with any standard or service (routine, customary or otherwise) related to the consulting, services, hardware or software industries.

AT&T DOES NOT WARRANT THAT THE INFORMATION IS ERROR-FREE. AT&T IS PROVIDING THE INFORMATION TO YOU "AS IS" AND "WITH ALL FAULTS." AT&T DOES NOT WARRANT, BY VIRTUE OF THIS DOCUMENT, OR BY ANY COURSE OF PERFORMANCE, COURSE OF DEALING, USAGE OF TRADE OR ANY COLLATERAL DOCUMENT HEREUNDER OR OTHERWISE, AND HEREBY EXPRESSLY DISCLAIMS, ANY REPRESENTATION OR WARRANTY OF ANY KIND WITH RESPECT TO THE INFORMATION, INCLUDING, WITHOUT LIMITATION, ANY REPRESENTATION OR WARRANTY OF DESIGN, PERFORMANCE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, OR ANY REPRESENTATION OR WARRANTY THAT THE INFORMATION IS APPLICABLE TO OR INTEROPERABLE WITH ANY SYSTEM, DATA, HARDWARE OR SOFTWARE OF ANY KIND. AT&T DISCLAIMS AND IN NO EVENT SHALL BE LIABLE FOR ANY LOSSES OR DAMAGES OF ANY KIND, WHETHER DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, SPECIAL OR EXEMPLARY, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, LOSS OF GOODWILL, COVER, TORTIOUS CONDUCT OR OTHER PECUNIARY LOSS, ARISING OUT OF OR IN ANY WAY RELATED TO THE PROVISION, NON-PROVISION, USE OR NON-USE OF THE INFORMATION, EVEN IF AT&T HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES OR DAMAGES.





Table of Contents

1	Introduction	1
2	Initializing the Index.....	2
2.1	Create Message Index.....	4
3	Managing Inactivity Timers.....	8





1 Introduction

This document is intended for developers who are using the In-App Messaging API to manage a user's message store. This document describes how an app can manage and maintain an index for the user.

This document describes how to set up the index and to initially sync up with the user's message store for boot strapping or after long periods of inactivity. In the case of steady state operations, the app should keep in sync with the message store by either using notifications or message polling. For information about polling, see the [In-App-Messaging API: Polling](#) document





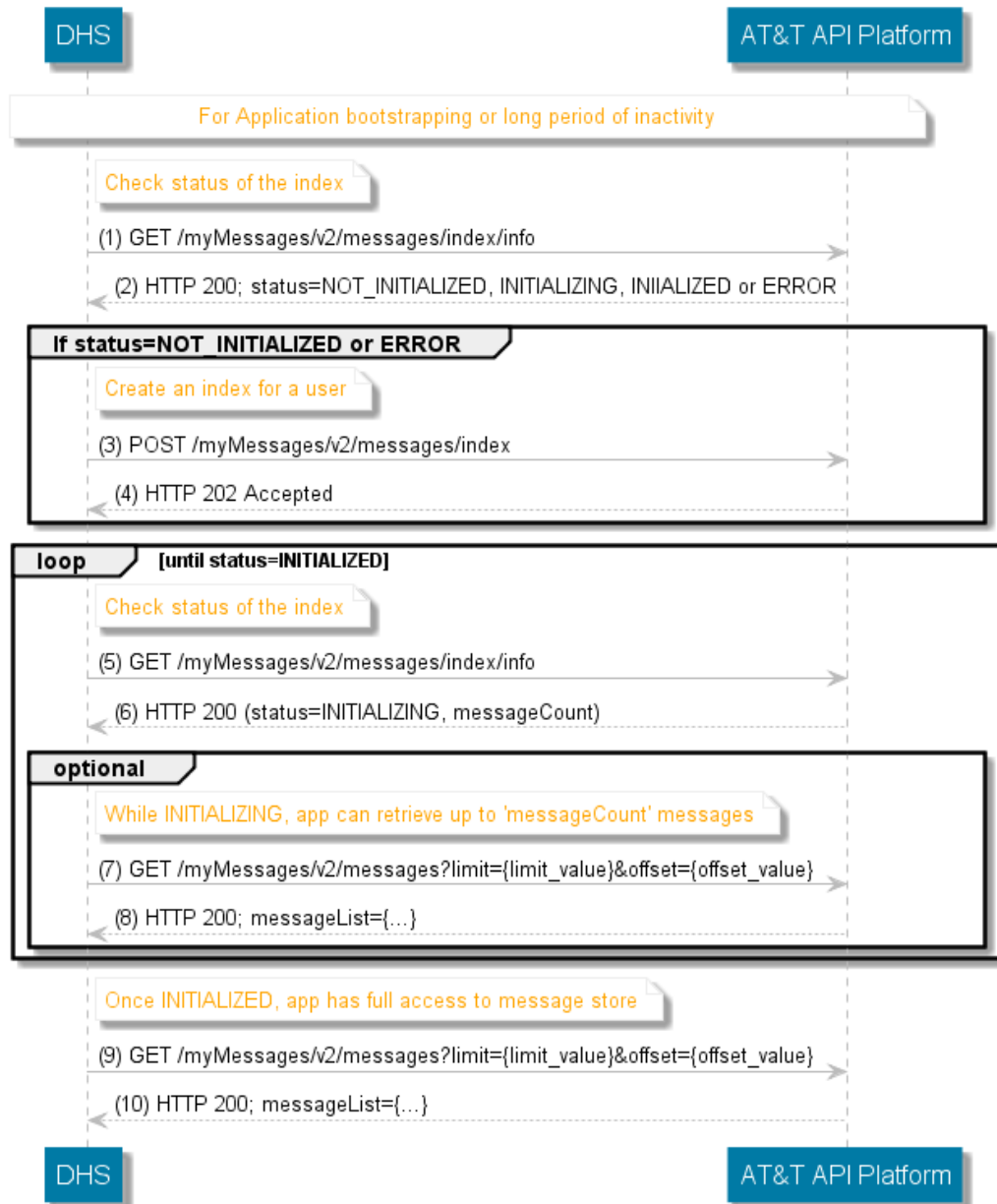
2 Initializing the Index

An index for each user must exist so that the app can interact with the message store for that user. The index should be initialized on initial startup of the app, if an error has occurred with the index, or if the app has been inactive for a period of time (see the next section, Managing Inactivity Timers). The following diagram illustrates the best practices for managing the index and retrieving the initial set of messages.





In-App Messaging Index Management





2.1 Create Message Index

The following procedure shows how to create an index for the user's message box.

1. Use the Get Message Index Info method to get the status of the index.

```
GET/myMessages/v2/messages/index/info HTTP/1.1
```

```
Host: api.att.com  
accept: application/json  
authorization: Bearer abcxyz123456
```

Example 2-1: Get the status of the index

2. The response to the Get Message Index Info operation will contain three parameters: status, state and messageCount.

The status parameter can have four possible values.

- ERROR - An error has occurred and the index should be re-created.
- INITIALIZED - The index has been created, and the message store can be accessed.
- INITIALIZING - A request to create the index has been sent, and the index is being created.
- NOT_INITIALIZED - The index has not yet been initialized and should be created.

The state parameter is used for retrieving updates to the message store.

The messageCount parameter indicates the number of messages that have been indexed. If the status parameter is set to INITIALIZING, then this parameter should increase with each subsequent Get Message Index Info request.





Note: If the messageCount parameter is not increasing with every Get Message Index Info request, then the process could be hanging and it may be necessary to perform another Create Message Index request.

```
HTTP /1.1 200 OK
Content-Type: application/json
Content-Length: 711

{
  "messageIndexInfo": {
    "status": "NOT-INITIALIZED",
    "state": "1388102635555",
    "messageCount": 0
  }
}
```

Example 2-2: Index status

3. If the status parameter is either NOT_INITIALIZED or ERROR, then a Create Message Index request should be made to create the index.

```
POST /myMessages/v2/messages/index HTTP/1.1
Host: api.att.com
accept: application/json
authorization: Bearer abcxyz123456
```

Example 2-3: Create Index if status is NOT_INITIALIZED or ERROR

4. If the request was successful, the response will be an HTTP Status Code 202, meaning that the index creation process has been initiated and the index is being created.
5. The Get Message Index Info method should be repeated to get the status of the index. This should continue as a loop until the status parameter is set to INITIALIZED. It is recommended that the app waits for at least 10 seconds between calls to the Get Message Index Info method.

There are two situations when the app should break the loop.

- An error is returned from the AT&T API Gateway.
- Initializing is taking too long. This is true if the messageCount parameter remains at the same count after a few Get Message Index Info requests.

In this case, the process should restart with step 3.

6. The Get Message Index Info response contains the status, state and message count. The messageCount parameter indicates the number of messages that are currently indexed.





7. As an option, while performing the Get Message Index Info loop, the app may retrieve messages up to the current message count value using pagination (limit and offset).

```
GET /myMessages/v2/messages?limit=500&offset=50 HTTP/1.1
Host: api.att.com
accept: application/json
authorization: Bearer abcxyz123456
```

Example 2-4: Retrieving messages using limit and offset parameters

8. The retrieved messages are returned.

```
HTTP /1.1 200 OK
Content-Type: application/json
Content-Length: 762
```

```
{
  "messageList": {
    "messages": [
      {
        "messageId": "r888",
        "from": {
          "value": "+14255551212"
        },
        "recipients": [
          {
            "value": "+12065551212"
          }
        ],
        "timestamp": "2014-11-18T19:05:17",
        "text": "Hello there!",
        "isFavorite": false,
        "isUnread": false,
        "type": "TEXT",
        "typeMetadata": {}
      }
    ],
    "offset": 0,
    "limit": 10,
    "total": 1,
    "state": "1416338570349",
    "cacheStatus": "INITIALIZED",
    "failedMessages": []
  }
}
```

Example 2-5: Retrieving messages





9. Once the status parameter is initialized, the app has full access to the user's message store and can fully utilize all of the In-App-Messaging methods. The retrieved messages are returned.





3 Managing Inactivity Timers

There are two inactivity timers to be aware of while managing the index.

- Once the index status is initialized, the index status is persistent for the user for a 30-day rolling period from when the Get Message Index Info request is sent, unless an error in the AT&T messaging system corrupts the index. If there has been no activity after 30 days, then the user's index will be removed entirely. To access the inbox again after the index has been removed, the index must be created again using the Create Message Index method.
- While the index status is initialized, the index is continuously synchronized with the source platform for 24 hours after a Get Message Index Info request is sent. This guarantees that a long-running app will continue to receive notifications for deltas in the customer's inbox. If the app is expected to run for periods longer than 24 hours, a new Get Message Index Info request should be sent at 24-hour intervals to reset the inactivity timer.

Note: Unnecessary Get Message Index Info requests should be avoided as well. In other words, the Get Message Index Info method should not be called at short intervals; every 24 hours is sufficient unless the method is needed for other purposes.

