Application Analytics

Integration Guide

Contents

CONTENTS 2
DOCUMENT CHANGE LOG 3
INTRODUCTION 4
BEFORE YOU START 4
HOW APPLICATION ANALYTICS WORKS 5
SETTING UP YOUR APPLICATIONID 5
SENDING ANALYTICS DATA TO BANGO 5
Posting the data 5
JSON Structure 6
Response to received batch 7
Event names 8
BATCH FIELDS 9
EVENT FIELDS 13
CUSTOM PARAMETERS 15
CAMPAIGN PARAMETERS 17

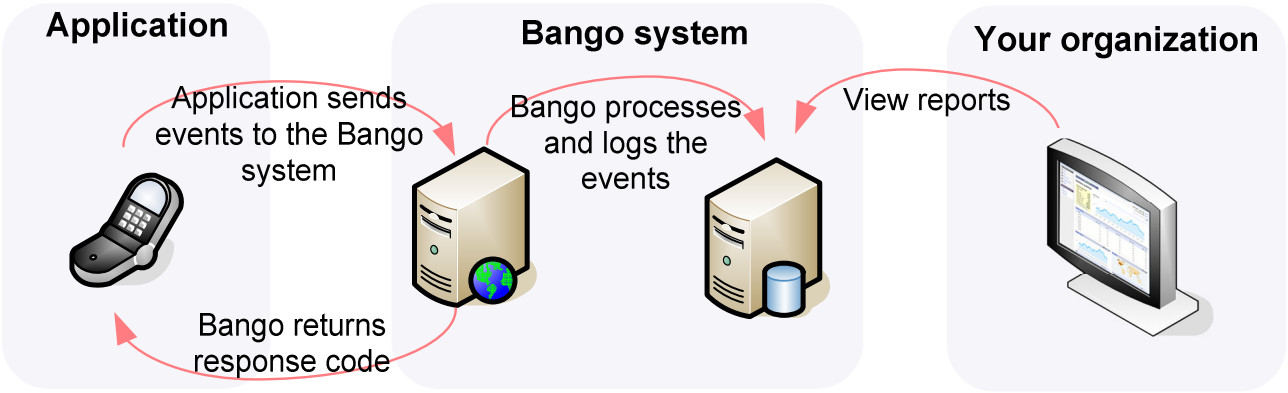
Document change log

Date	Version	Change detail
1st September 2010	1.0	Published
7th October 2010	1.0.1	Corrected example customer parameter to BGO_CUSTOM1 (incorrectly specified as BGO_CUSTOMPARAMETER1) Corrected BGO_VALUECUR and BGO_COSTCUR parameter names – incorrectly specified with two 'R's.

Introduction

Bango Application Analytics is an analytics solution for mobile applications. When an event happens in a mobile application it is sent to Bango. Bango will then store this data and it will be available to be reported on. This document provides detail on integrating your mobile application with our recording servers. We do provide a set of SDK's that perform this job for you. Please note the Bango Analytics integration is for use by those with significant development experience. Please see the Bango Management Tools and Bango Knowledgebase for more information.

This diagram shows the Application Analytics architecture:



If you have chosen to integrate with our open recording API yourself then you will be responsible for collecting the data from the mobile application/device and transmitting through the internet to Bango.

Before you start

It is recommended that you use a separate Analytics account for integration testing so that your test data is separate from your live data. Once the integration is ready to go live you can switch to your live account.

Please contact your account manager or Bango customer services for information or sign up for a one month free trial to get started at Bango.com.

How Application analytics works

Setting up your ApplicationId

Each application that you wish to use Bango analytics for needs an ApplicationId. An ApplicationId is a simply a unique reference for your application. If your application is distributed on different platforms you have a choice with how to use the ApplicationId:

1. Use a single ApplicationId for all versions of the application across all platforms. This will mean that the data from all versions of the application will be aggregated for display in the app analytics reports. You can filter the reports by OS version to see data from individual apps.
2. Use a unique ApplicationId for every different version of the application. This will mean that the data from each version of the platform will be displayed separately in the app analytics reports. This gives you the highest level of granularity for your data.

To set up an ApplicationId visit the 'Manage' section of the Bango.com Management Tools and click on 'Analytics set-up' in the left menu then click on 'App analytics'.

Sending analytics data to Bango

Once an event has occurred in a mobile application it can be sent straight away or stored in the mobile device's memory to be sent at a later time. Bango has a concept of batches and events. You can send one batch with each transmission and each batch may contain many events. This model suits the storing of events on the mobile device's memory ready for sending as a batch when the application closes. This means that the sending of event data does not create any network traffic when the application is in use. This model is less network intensive because data common to each event is only sent once with each batch.

Posting the data

The transmission of the batch data is done using a HTTP POST (which can be secured using SSL) to one of the URLs below -

Live URL:

http(s)://bango.net/appanalytics/

Live URL with debugging information:

http(s)://bango.net/appanalytics/debug/

Please note that the 'debug' URL will log events the same as the non-debug URL. The only difference is the extended debugging information in the response.

The JSON should be sent as the POST data. The Content-Type should be '*text/x-json*'

Security

We strongly recommend that you connect to our URL using SSL when performing the HTTP POST. This means that the data is secure while being sent to Bango's servers.

Example post data:

{"Events":[{"IsValid":true,"EventParameters":[],"EventName":"BGO_APP_SESSION_START","EventValue":"","EventTime":"2010-09-23T14:51:56.156","EventLocalTime":"2010-09-23T14:51:56.156","ConnectionType":"WIFI"},{"IsValid":true,"EventParameters":[{"Name":"BGO_CUSTOM1","Value":"LEVEL_6"}],"EventName":"BGO_APP_SESSION_END","EventTime":"2010-09-23T14:51:56.156","EventLocalTime":"2010-09-23T14:51:56.156","ConnectionType":"WIFI","OperatorConnectionType":""}],"ApplicationId":"11188812345","UserId":"12345","SessionRestartTime":10,"IP":"","DeviceIsMobile":false,"ConnectionType":"WIFI","IsRoaming":false}

JSON Structure

The format that is used for the transmission of the batch is JSON (JavaScript Object Notation). This provides a lightweight text-based way of formatting the batch. For more information on JSON please see (http://www.json.org/)

An example of the JSON batch:


```
{
  "ApplicationId":"11188812345",
  "UserId":"123456",
  "SessionRestartTime":10,
  "ConnectionType":"WIFI",
  "Events":
        [
            {
            "EventName":"BGO_APP_SESSION_START",
            "EventDetail":"",
            "EventTime":"2010-09-23T14:51:56.156",
            "EventLocalTime":"2010-09-23T13:51:56.156",
            "ConnectionType":"WIFI",
            "EventParameters":
                [
                    {
                        "Name":"BGO_CUSTOM1",
                        "Value":"LEVEL_5"
                    }
                ]
            },
            {
            "EventName":"BGO_APP_SESSION_END",
            "EventDetail":"",
            "EventTime":"2010-09-23T14:52:55",
            "EventLocalTime":"2010-09-23T13:52:55",
            "ConnectionType":"WIFI",
            "EventParameters":
                [
                    {
                        "Name":"BGO_CUSTOM1",
                        "Value":"LEVEL_6"
                    }
                ]
            }
        ]
}
```


The fields that relate to the entire batch

A single event

Fields relating to the event

Extra parameters relating to event

The batch is the root element. The fields on the batch are fields that apply to every event. For example the ApplicationId and the UserId will not change between events, so there is little point including them with each event. As a result they are batch level fields.

EventName on the other hand will change between each event and as a result it is an event level field.

The event(s) are the elements of the 'Events' array which is a batch level field

"Events":[{ *event 1* } , { *event 2* } , { *event 3* }]

The event parameters are elements of the 'EventParameters' array which is a event level field

"EventParameters": [{ *event parameter 1* } , { *event parameter 2* }]

Each event parameter is an object with a 'Name' and a 'Value'

{ "Name":" *parameter name* " , "Value":" *parameter value* " }

The parameter name must be one of the pre-defined custom parameter names (see page 15 of this guide).

Response to received batch

When we receive a HTTP POST we will give you an instant text response code in the response data. Please note that this is not the HTTP response code, this will always be 200 (OK).

For all the possible response codes, please see the following table:

Response code	Description	Corrective action
OK	The batch has been successfully processed	N/A
OK_WITH_ERRORS	Some events and/or batch fields may have failed validation and as a result a validation error has been logged. However not all events failed to be recorded.	Check the validation error log in the Management Tools and take corrective action.
DATA_MISSING	There was some data missing from the batch which meant that the batch could not be processed or There is data missing from all the events, which meant that no events could be logged.	Make sure that all the required fields have a value
POST_DATA_MISSING	The post data is missing from the HTTP POST request	Ensure that you are sending the JSON in the requests POST data.
DATA_INVALID	The JSON data sent could not be parsed	Ensure that the JSON is valid and URL encoded
BLOCKED	Your ApplicationId has been blocked from logging data	Contact Bango customer services
UNKNOWN_ERROR	There has been an unknown error	If this persists, contact Bango customer services

If you receive a response that you are not expecting, you can find out more information about reason for the response if you use the 'with debugging' URL.

Example response from the LIVE URL:

```
DATA_MISSING
```

Example response from the LIVE URL with debugging:

```
DATA_MISSING
ERRORS:
Description:'ApplicationId' is missing from batch          Severity:Major      Level:Batch
```

Event names

Each event that is contained within a batch must have a name. You can pick any name you wish for each event. You could for example have a 'LEVEL_COMPLETE' event if your application was a game and you wish record when a user finishes a level.

Bango has provided some predefined event names that you can use. If you want to make full use of the rich reporting features of Bango Analytics then you should implement these event names on the relevant events. For example, the sessions report will not work unless you have used the predefined session event names.

Here is a full list of the pre-defined event names:

Event name	Purpose
BGO_APP_SESSION_START	The start of a session. This event name should be used when the user opens the app.
BGO_APP_SESSION_END	The end of a session. This event name should be used when the user closes the app.
BGO_APP_IDLE_START	This event name should be used when the application looses focus
BGO_APP_IDLE_END	This event name should be used when the application re-gains focus
BGO_APP_VIDEO_START	This event name should be used at the start of a video play (note the video title should be passed in the eventDetail field)
BGO_APP_VIDEO_STOP	This event name should be used at the end of a video play (note the video title should be passed in the eventDetail field)
BGO_APP_ERROR	This event name should be used when an error has occurred in the application
BGO_APP_PAGEVIEW	This event name should be used when a user views a page in you application such as a news article
BGO_APP_SEARCH	This event name should be used when a search has been initiated inside your application

Batch fields

Complete list of fields that can be sent on the batch (root) node:

Field name	Data type	Required	Description
ApplicationId	String(50)	Yes	The ApplicationId that has been set up for the application using the Management Tools **Example:** *"ApplicationId":"11188800000"*
UserId	String(50)	Yes	The unique identifier that you have for the user of the application. For example, this could be an email or a GUID or just a numerical value **Example:** *"UserId":"12345"*
SessionRestartTime	Int32	Yes	The period in seconds, that after we receive a 'BGO_APP_SESSION_END' event we will still count events in that previous session **Example:** *"SessionRestartTime":10*
IP	String(15)	No	The internet address of the device. You can, if you wish supply the IP of the device. If you do not supply this field we will instead use the IP of the device when we receive the data **Example:** *"IP":"123.123.123.123"*
ApplicationType	String(50)	No	The type of the application. For example, Lite, Beta, Full, Free **Example:** *"ApplicationType":"Lite"*
ApplicationChannel	String(50)	No	Typically how the app was distributed. For example, App store, Marketplace **Example:** *"ApplicationChannel":"AppStore"*
ApplicationVersion	String(50)	No	The version of the application **Example:** *ApplicationVersion":"1.0"*
CountryISO	String(3)	No	The ISO3 country code for the country of the device. For example, USA, GBR If you cannot obtain a three character country code then we can accept a two character code instead. **Example:** *"CountryISO":"USA"*
OperatorName	String(20)	No	The name of the operator that the device uses. **Example:** *"OperatorName":"Sprint"*
DeviceBrand	String(100)	No	The brand of the device. For example, Sprint, Verizon **Example:** *"DeviceBrand":"Sprint"*
DeviceMake	String(100)	No	The make of the device. For example, HTC, Apple

DeviceModel	String(100)	No	**Example:** *"DeviceMake":"Apple"* The model of the device. For example, Desire, iPhone 4 **Example:** *"DeviceModel":"Desire"*
DeviceVersion	String(50)	No	The version of the device. For example, 1.1 **Example:** *"DeviceVersion":"1.1"*
DeviceIsMobile	Bool	No	Is the device a mobile device **Example:** *"DeviceIsMobile":true* *Must be either **true** or **false***
OperatingSystem	String(50)	No	The name of the operating system. For example, Android **Example:** *"OperatingSystem":"Android"*
OperatingSystemVersion	String(50)	No	The version of the operating system. For example, 1.1 **Example:** *"OperatingSystemVersion":"1.1"*
OperatingSystemVersion Name	String(50)	No	The name of the version. For example, cupcake **Example:** *"OperatingSystemVersionName":"CUPCAKE"*
LocationX	Decimal	No	The latitude of the devices location **Example:** *"LocationX":36.9525000*
LocationY	Decimal	No	The longitude of the devices location **Example:** *"LocationY":-110.0725000*
LocationZ	Decimal	No	The altitude of the devices location (in meters) **Example:** *"LocationZ":36.5*
LocationAccuracy	Decimal	No	The accuracy as a decimal of the devices location **Example:** *"LocationAccuracy":10.0*

<table>
<tr><td>LocationSource</td><td>String(20)</td><td>No</td><td>The source of the location data. Must be one of the following:<table><tr><td>GPS</td><td>The location was obtained using the Global Positioning System</td></tr><tr><td>WIFI</td><td>The location was obtained using WiFi hot spots</td></tr><tr><td>NETWORK</td><td>The location was obtained using cell positioning</td></tr><tr><td>IP</td><td>The location was obtained using the devices IP address</td></tr><tr><td>OTHER</td><td>The location was obtained using some other means</td></tr></table>Example:
"LocationSource":"GPS"</td></tr>
<tr><td>LocationTown</td><td>String(100)</td><td>No</td><td>The town closest to the devices location
Example:
"LocationTown":"Fayetteville"</td></tr>
<tr><td>LocationRegion</td><td>String(100)</td><td>No</td><td>The region of the devices location
Example:
"LocationRegion":"North Carolina"</td></tr>
<tr><td>LocationCountryISO</td><td>String(3)</td><td>No</td><td>The ISO3 country code for the devices location
Example:
"LocationCountryISO":"USA"</td></tr>
<tr><td>ConnectionType</td><td>String(20)</td><td>No</td><td>The current connection type.
Must be one of the following:<table><tr><td>WIFI</td><td>The device is connected using WIFI</td></tr><tr><td>OPERATOR</td><td>The device is connected using it’s operator connection</td></tr><tr><td>OTHER</td><td>The device is connected by some other means</td></tr></table>Example:
“ConnectionType”:”WIFI”</td></tr>
<tr><td>OperatorConnectionType</td><td>String(20)</td><td>No</td><td>The wireless standard that the mobile connection uses<table><tr><td>GPRS</td><td>Operator connection is GPRS</td></tr><tr><td>EDGE</td><td>Operator connection is EDGE</td></tr><tr><td>UMTS</td><td>Operator connection is UMTS</td></tr><tr><td>HSDPA</td><td>Operator connection is HSDPA</td></tr><tr><td>HSUPA</td><td>Operator connection is HSUPA</td></tr><tr><td>HSPA</td><td>Operator connection is HSPA</td></tr><tr><td>CDMA</td><td>Operator connection is CDMA</td></tr><tr><td>EVDO_0</td><td>Operator connection is EVDO revision 0</td></tr><tr><td>EVDO_A</td><td>Operator connection is EVDO revision A</td></tr><tr><td>OTHER</td><td>Some other connection type</td></tr></table>Example:
"OperatorConnectionType":"UMTS"</td></tr>
</table>

IsRoaming	Bool	No	Is the device on a location other than its 'home' area
			Example: *"IsRoaming":true* *Must be either **true** or **false***
Events	Array	Yes	The array that contains the event objects
			Example: *"Events":[{ event 1 } , { event 2 } , { event 3 }]*

Event fields

Field name	Data type	Required	Description
EventName	String(50)	Yes	The event name is identifier for the type of event that you would like to record. You can create your own event names simply sending it in this field. Bango has a set of predefined event names. It is highly recommended that you use these pre-defined event names on the relevant event in order to get the best use of Bango's rich reporting tools. You can get a full list of event names see the event names section **Example:** *"EventName":"BGO_APP_SESSION_START"*
EventDetail	String(200)	No	Any extra detail that you would like to attach to the event **Example:** *"EventDetail": "User viewed the review section"*
EventValue	String(50)	No	This is a value that you can associate with the event **Example:** *"EventValue":"ASBF54D"*
EventTime	DateTime	Yes	The time that the event occurred in UTC *Must be in the following format :"yyyy-MM-dd'T'HH:mm:ss.SSS"* **Example:** *"EventTime":"2010-08-25T15:20:44.134"*
EventLocalTime	DateTime	No	The time that the event occurred in the local time zone *Must be in the following format :"yyyy-MM-dd'T'HH:mm:ss.SSS"* **Example:** *"EventLocalTime":"2010-08-25T14:20:44.134"*
ConnectionType	String(20)	No	The current connection type. Must be one of the following: WIFI – The device is connected OPERATOR – The device is connected using its operator connection OTHER – The device is connected by some other means **Example:** *"ConnectionType":"WIFI"*
OperatorConnectionType	String(20)	No	The wireless standard that the mobile connection uses. Must be one of the following: GPRS – Operator connection is GPRS EDGE – Operator connection is EDGE UMTS – Operator connection is UMTS HSDPA – Operator connection is HSDPA HSUPA – Operator connection is HSUPA HSPA – Operator connection is HSPA CDMA – Operator connection is CDMA EVDO_0 – Operator connection is EVDO revision 0 EVDO_A – Operator connection is EVDO

			revision A OTHER: Some other connection type **Example:** *"OperatorConnectionType":"UMTS"*
ErrorId	String(50)	No	If this event relates to an error in the application, then an errorId can be attached using this field **Example:** *"ErrorId":"ERROR_1"*
ErrorLevel	Int32	No	This field indicates to Bango where the error originated. Sending the value of '1' will indicate that it was an error in the application. **Example:** *"ErrorId" 1*
EventParameters	Array	yes	This is the array that contains any event parameters that you wish to attach to the event **Example:** *"EventParameters":[* *{* *"Name":"BGO_CUSTOM1",* *"Value":"LEVEL_6"* *}* *]* **Or if you don't wish to send any parameters, supply an empty array:** *"EventParameters":[ ]*

Custom parameters

These are the custom parameters that can be attached to an event.

Bango parameter	Recommended use	Maximum size
bgo_region	The region parameter can be used to indicate the regional variation of an app or as a geographical indicator.	200 characters
bgo_section	The section parameter should be used to record which section of an app the event was generated in e.g. News, Finance, Sport. Use in conjunction with bgo_subsection to give a finer level of granularity.	200 characters
bgo_subsection	The subsection parameter should be used to define a sub-section of an app (e.g. US News, UK News, World News). Use in conjunction with bgo_section to give a greater level of detail.	200 characters
bgo_media	The media parameter should be used to define the media type being analysed (e.g. video, ringtone, audio). Use in conjunction with bgo_author to give a greater level of detail.	200 characters
bgo_author	The author parameter should be used to indicate the author of content. Use in conjunction with bgo_media to give a greater level of detail.	200 characters
bgo_gender	The gender parameter should be used to indicate the gender of the user. Used to record male or female, any other value will be ignored.	M, MALE, F or FEMALE
bgo_age	The age parameter should be used to indicate the age of a user. The following ages are mapped to a single letter: < 16 yrs: A 16 - 20 yrs: B 21 - 25 yrs: C 26 - 30 yrs: D 31 - 35 yrs: E 36 - 40 yrs: F 41 - 50 yrs: G 51 - 60 yrs: H 60 + yrs: I	A, B, C, D, E, F, G, H or I, any other value will be ignored.
bgo_identity	The identity parameter should be used to record an identity you have for a user. This is separate from the Bango User ID and	200 characters

	should be used for an identity you might have for the user.	
bgo_value	The value parameter should be used to define a monetary value associated with an event. Can be used in conjunction with bgo_valuecur to give a greater level of detail.	200 characters
bgo_valuecur	The value currency parameter should be used to define the currency (code, symbol or description) for the value defined above. To be used in conjunction with bgo_value.	10 characters
bgo_cost	The cost parameter should be used to define a cost associated with an event. Can be used in conjunction with bgo_costcur to give a greater level of detail.	200 characters
bgo_costcur	The cost currency parameter should be used to define the currency (code, symbol or description) for the cost defined above. To be used in conjunction with bgo_cost.	10 characters
bgo_custom1	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom2	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom3	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom4	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom5	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom6	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom7	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom8	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom9	To be used to record a custom parameter, at reporting stage you will be able to map meaning to each custom parameter	200 characters
bgo_custom10	To be used to record a custom parameter, at reporting stage you will be able to map	200 characters

	meaning to each custom parameter	
bgo_videotype	The video type parameter should be used to indicate if the video was live or on-demand	LIVE, ONDEMAND

Campaign parameters

These are parameters that are used to track campaign information across web pages. If you want to send campaign information out from your application these parameters can be used. For a full guide please see http://bango.custhelp.com/app/answers/detail/a_id/1189. You can also use these parameters within your application to associate information with events.

Bango parameter	Recommended use	Maximum size
bgo_campaign	Use to track the different advertising campaigns or product promotions that your business creates. An example of this would be a mobile content store tracking the effectiveness of their “Hot Summer Ringtone Sale” advertising campaign.	200 characters
bgo_source	Visitors to your website must come from somewhere. That is, each referral to a website has an origin, or Source . Examples of sources are the Google search engine, an email newsletter, or a referring web site. There may be several Sources for each campaign. For example, the “Hot Summer Ringtone Sale” is advertised in both an email newsletter and a banner ad. In this case, both “newsletter” and “banner ad” would be possible Sources.	200 characters
bgo_medium	The Medium helps to qualify the source. Together, the Source and Medium provide specific information about the origin of a referral. For example, if the Source is “Google,” the medium might be “CPC,” indicating a sponsored link. Or, the Medium might be “organic,” indicating a link in the unpaid search results. In the case of a “newsletter” Source, examples of Medium include “email” and “print.”	200 characters
bgo_content	The Content variable indicates the version of an ad on which a visitor clicked. Labelling your content versions allows you to determine which one is most effective at attracting profitable leads. For example, if you had two versions of a banner ad, you could use the Content parameter to identify which one is bringing more visits to	200 characters

	your site.	
bgo_term	The Term is a keyword or phrase that matches what a user types into a search engine. For example, a link in a cost-per-click (CPC) ad would be tagged with the Term that triggered the ad. In our example, the Term might be “hot ringtones.”	200 characters
bgo_owner	The Owner parameter can be used to assign an individual, group or organisation responsibility for a campaign link.	200 characters