CRICHQ

CricEngine API V1.1.2

Outputting data for external devices

Creator: Daniel Ryan

Date created: 27/6/16

Introduction

This technical document will explain the API that allows external devices to get information from the CricHQ mobile application. This API has been designed with scoreboards in mind. It will allow external devices to connect to a mobile device with CricHQ installed and receive notification events. External devices will listen to these notification events and use this information to work out what screen to show. The external device will call the necessary APIs to get the information it needs for its view. The API is versioned to allow future updates. The information passed between devices will be all in JSON.

Setting Up

The mobile device will connect to the external device using Bonjour. Connection to our device will be done by a simple APP UI that shows all the listed external devices, allowing the user to make the choice. Also, we will support auto connection if enabled in the setting, this will be enabled by default. The auto connection will connect to the first device that replies back.

The external device should give a good service name that the user will recognise e.g Scoreboard_9645.

Bonjour settings:
Service Port: 0 (Any)
Service Type: _cricviewing
Service Protocol: _tcp
Service Domain: local

The service type will be registered here in the future: http://www.iana.org/form/ports-services

Length of Packet

At the start of each block of Json, there will be a length (little-endian) value. The length will be 8 bytes and this number gives you the Json length. When sending Json you will also need to include this 8-byte length value.

Security

No security support yet. Anyone accessing the network with the CricHQ application could take

over the connected device (e.g scoreboard).
A passcode could be typed in for protected external devices as a future feature.

Versioning

The API (for a particular version) will not change over time unless it is totally necessary, e.g to fix bugs. Statistics calculated like MVP, DLS, projections, etc values will change over time and API versions will have no effect.

Version	Major Changes
1.0.0	First Version.
1.1.1	Add the method “getStatsToWin”. Add the method “getLast10Events”. Event notifications now have guid and timestamp information. Rename JSON keys: -“foursCount” to “boundaryFourCount”. -“sixesCount”, to “boundarySixCount“. -“partnershipBalls“, to “partnershipDeliveries“. -“legalDeliveriesBowledIndex“ to “legalDeliveriesBowled”. Add “showTargetResults” flag for DLS. Add “ignoreStats” flag and “legalDeliveryDisplay” for actions. Add match guid to event notifications.
1.1.2	Added “clientTimestamp” to event notifications. Added “order” to event notifications.

Application Programming Interface

Once the external device is connected to the scoring device, it can call any method by passing the JSON to the scoring device. The API will auto fire notifications of events back to the external device with the event type. The external device can call any method in the API.

To grab data from a mobile device running CricHQ you always need to send JSON to say which method (as `method)` are you calling and what version (as `apiVersion`) of the method.

The returned JSON always gives back the method name (under `methodCaller`) that was originally called and the API version number (as `apiVersion`) .

Input Arguments	Type	What does it do

`method`	`String`	Method being called.
`apiVersion`	`String`	Api version to be used.

Output Arguments	**Type**	**What does it do**
`methodCaller`	`String`	Method called.
`apiVersion`	`String`	Api version used.

Basic example sent:

```
{
      "method": "getScorecard",    //Method name to call.
      "apiVersion": "1.0.0"        //API version to use for this method.
}
```

Basic example received:

```
{
      "methodCaller": "getScorecard",          //Method name that was called
      "apiVersion": "1.0.0"        //API version which was used for this method.
}
```

API methods list

Method name	**What does it do**
getLatestVersion	Get the latest version of the API.
getScorecard	Get the stats needed to build a full scorecard view.
getLastActionCommentary	Get last game action commentary.
getBattingBowlingView	Get the stats needed to build a batting and bowling view.
getOverCommentary	Get commentary for an over.
getBattingTeamAnalysis	Get batting team analysis stats.
getOverSummary	Get over summary.
getDuckworthLewisStern	Get Duckworth Lewis Stern (DLS) stats (latest version is calculated). Previously called Duckworth Lewis.

getOversLost	Get the overs lost in a match.
getMatchState	Get the state of the match.
getEventCount	Get event count.
getOverIndex	Get active over index.
getInningsIndex	Get active innings index.
getMatchScoreView	Get the current scores to build a score view.
getTickerTape	Get the text shown on the ticker tape.
getEventTypesPlayed	Get a list of all the event types were played in the match.
getLastOverSummary	Get last over summary.
getStatsToWin	Get to win stats.

API methods

Get Latest Version

Method name: `getLatestVersion`
Get the latest version of the API.

Output Arguments	**Type**	**What does it do**
`latestVersion`	`String`	Latest API version.

Example Json sent:

```
{
	"method": "latestVersion",
	"apiVersion": "1.0.0"
}
```

This could return:

```
{
	"methodCaller": "latestVersion",
	"apiVersion": "1.0.0",
	"latestVersion": "1.0.5"
}
```

Get Scorecard

Method name: `getScorecard`
Get the stats needed to build a full scorecard view.

Output Arguments	Type	What does it do
`awayTeamBasic`	`Object (Team Basic)`	Away team.
`inningsIndex`	`int`	Innings index.
`homeTeamBasic`	`Object (Team Basic)`	Home team.
`inningsScorecards`	`Array of Objects (Innings Scorecard)`	Innings scorecards.
`isHomeTeamBatting`	`Boolean`	Is the home team batting.
`matchState`	`Enum (Match State)`	The state of what the match is in.
`matchEndCategory`	`Enum (Match End Category)`	Able to check if there is a winner at the end of a match.

Example Json sent:

```
{
      "method": "getScorecard",
      "apiVersion": "1.0.0"
}
```

This could return:

```
{
  "apiVersion" :"1.0.0",
  "awayTeamBasic" : {
     "clubServerId" :"51785",
     "id" : 2,
     "name" :"Sunrisers Hyderabad, SH",
     "serverId" :"51806"
   },
  "homeTeamBasic" : {
     "clubServerId" :"51777",
     "id" : 1,
     "name" :"Royal Challengers Bangalore, RCB",
     "serverId" :"51803"
   },
  "inningsIndex" : 0,
  "inningsScorecards" : [
      {
        "batterListStats" : [
            {
              "batter" : {
                 "id" : 4,
                 "name" :"AB de Villiers",
                 "serverId" :"283428",
                 "unsure" : false
               },
              "batterOrderIndex" : 2,
              "deliveriesFaced" : 1,
              "dotBallCount" : 0,
              "foursCount" : 0,
              "isDismissed" : false,
              "overCount" : 0,
              "score" : 1,
              "sixesCount" : 0,
              "strikeRate" : 100,
              "timeInPlay" : -51
            },
            {
              "batter" : {
                 "id" : 2,
                 "name" :"M Bisla",
                 "serverId" :"780743",
                 "unsure" : false
               },
              "batterOrderIndex" : 1,
              "deliveriesFaced" : 6,
              "dotBallCount" : 2,
              "foursCount" : 0,
              "isDismissed" : true,
              "outAction" : {
                 "actionType" : 3,
                 "bowler" : {
                    "id" : 12,
                    "name" :"B Kumar",
     "serverId" :"365001",
     "unsure" : false
   },
  "extraPoints" : 0,
  "extraType" : 0,
  "facingBatsman" : {
     "id" : 2,
     "name" :"M Bisla",
     "serverId" :"780743",
     "unsure" : false
   },
  "fielders" : null,
  "id" : 18,
  "inningsIndex" : 0,
  "keeper" : {
     "id" : 20,
     "name" :"NV Ojha",
     "serverId" :"419333",
     "unsure" : false
   },
  "legalDeliveryIndex" : 5,
  "nonActiveBowler" : {
     "id" : 21,
     "name" :"RS Bopara",
     "serverId" :"1032691",
     "unsure" : false
   },
  "nonFacingBatsman" : {
     "id" : 1,
     "name" :"CH Gayle",
     "serverId" :"284104",
     "unsure" : false
   },
  "outPlayer" : {
     "id" : 2,
     "name" :"M Bisla",
     "serverId" :"780743",
     "unsure" : false
   },
  "overIndex" : 2,
  "wicketType" : 3
},
      "overCount" : 3,
      "score" : 3,
      "sixesCount" : 0,
      "strikeRate" : 50,
      "timeInPlay" : 41
    },
    {
      "batter" : {
         "id" : 1,
         "name" :"CH Gayle",
         "serverId" :"284104",
         "unsure" : false
       },
      "batterOrderIndex" : 0,
      "deliveriesFaced" : 15,
      "dotBallCount" : 1,
      "foursCount" : 4,
      "isDismissed" : false,
      "overCount" : 3,
      "score" : 41,
      "sixesCount" : 3,
      "strikeRate" : 273.33334350585938,
      "timeInPlay" : 0
    }
 ],
"bowlerListStats" : [
    {
      "average" : 0,
      "bowler" : {
         "id" : 21,
         "name" :"RS Bopara",
         "serverId" :"1032691",
         "unsure" : false
       },
      "legalDeliveriesBowled" : 4,
      "legalDeliveriesBowledCount" : 10,
      "legalDeliveriesBowledIncompleteOversCount" : 0,
      "dotBallsCount" : 1,
      "maidens" : 0,
      "noBallPoints" : 0,
      "overCount" : 1,
      "rpo" : 15,
      "score" : 25,
      "wicketsCount" : 0,
      "widePoints" : 0
    },
    {
      "average" : 20,
      "bowler" : {
         "id" : 12,
         "name" :"B Kumar",
         "serverId" :"365001",
         "unsure" : false
       },
      "legalDeliveriesBowled" : 0,
      "legalDeliveriesBowledCount" : 12,
      "legalDeliveriesBowledIncompleteOversCount" : 0,
      "dotBallsCount" : 2,
      "maidens" : 0,
      "noBallPoints" : 0,
      "overCount" : 2,
      "rpo" : 10,
      "score" : 20,
      "wicketsCount" : 1,
      "widePoints" : 0
    }
 ],
"notes" : null,
"powerPlays" : null,
"stats" : {
   "actionIndex" : 4,
   "boundary6Count" : 3,
   "byePoints" : 0,
   "byesCount" : 0,
   "deliveryCount" : 22,
   "dotBallPercent" : 13.636363983154297,
   "dotBallsCount" : 3,
   "extraPoints" : 0,
   "legByePoints" : 0,
   "legByesCount" : 0,
   "legalDeliveries" : 4,
   "maidenCount" : 0,
   "maidenPercent" : 0,
   "noBallPoints" : 0,
           "noBallsCount" : 0,
           "overIndex" : 3,
           "partnershipDeliveries" : 4,
           "partnershipScore" : 12,
           "penaltyBattingPoints" : 0,
           "penaltyBowlingPoints" : 0,
           "rpo" : 12.272727012634277,
           "runsFromBoundariesCount" : 34,
           "runsFromBoundariesPercent" : 75.555557250976562,
           "score" : 45,
           "wicketCount" : 1,
           "wideAndNoBallsPercent" : 0,
           "widePoints" : 0,
           "widesCount" : 0
         }
      }
   ],
  "isHomeTeamBatting" : true,
  "matchEndCategory" : -1,
  "matchState" : 2,
  "methodCaller" :"getScorecard"
}
```

Get Last Action Commentary

Method: `getLastActionCommentary`
Get last game action commentary.

Object: Commentary Action

Example Json sent:

```
{
      "method": "getLastActionCommentary",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "actionType":3,
   "apiVersion":"1.0.0",
   "bowler":{
      "id":9,
      "name":"MA Starc",
      "serverId":"282770",
      "unsure":false
   },
   "commentary":"[bowler] to [batsman], Wicket! [outbatsman] 0. Big
appeal for LBW - That's out!",
   "facingBatsman":{
      "id":12,
      "name":"B Kumar",
      "serverId":"365001",
      "unsure":false
   },
   "historicalStats":{
      "score":173,
      "wicketCount":8
   },
   "id":241,
   "legalDeliveryIndex":5,
   "methodCaller":"getLastActionCommentary",
   "overIndex":19
}
```

Get Over Summary

Get the main over ball display.
Method name: `getOverSummary`

Output Arguments	Type	What does it do
`actions`	`Array of Objects (Action)`	Ball actions in this over.
`inningsIndex`	`int`	Innings index that this over is in.
`isActive`	`Boolean`	Is this over active.
`isComplete`	`Boolean`	Is this over complete.
`isMaiden`	`Boolean`	Is this over a maiden.
`deliveries`	`int`	The number of deliveries in this over.
`legalDeliveries`	`int`	The legal number of deliveries in this

		over.
`id`	`int`	The ID of the over.
`index`	`int`	The index of the over.
`score`	`int`	The score for this over.
`remainingDeliveries`	`int`	Remaining deliveries in this over.

Example Json sent:

```
{
      "method": "getOverSummary",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "actions":[
      {
         "actionType":1,
         "boundaryType":2,
         "bowler":{
            "id":21,
            "name":"RS Bopara",
            "serverId":"1032691",
            "unsure":false
         },
         "facingBatsman":{
            "id":1,
            "name":"CH Gayle",
            "serverId":"284104",
            "unsure":false
         },
         "hasPitchMap":false,
         "hasWagonWheel":false,
         "id":19,
         "ignoreStats":false,
         "inningsIndex":0,
         "keeper":{
            "id":20,
            "name":"NV Ojha",
         "serverId":"419333",
         "unsure":false
      },
      "legalDeliveryDisplay":1,
      "legalDeliveryIndex":0,
      "nonActiveBowler":{
         "id":12,
         "name":"B Kumar",
         "serverId":"365001",
         "unsure":false
      },
      "nonFacingBatsman":{
         "id":4,
         "name":"AB de Villiers",
         "serverId":"283428",
         "unsure":false
      },
      "overIndex":3,
      "pitchMapBackSquare":0,
      "pitchMapBadBowl":false,
      "pitchMapPitchSquare":0,
      "runs":6,
      "toKeeper":false,
      "wagonWheelCutProportion":0,
      "wagonWheelEllipseProportion":0,
      "wagonWheelHeightProportion":0,
      "wagonWheelHitAngleRadians":0
   },
   [...]
],
"apiVersion":"1.1.0",
"deliveries":4,
"id":4,
"index":3,
"inningsIndex":0,
"isActive":true,
"isComplete":false,
"isMaiden":false,
"legalDeliveries":4,
"methodCaller":"getOverSummary",
"remainingDeliveries":2,
"score":12
```

```
}
```

Get Last Over Summary

Get the last over summary.
Method name: `getLastOverSummary`

Same as getOverSummary.

Get Over Commentary

Method name: `getOverCommentary`

Output Arguments	**Type**	**What does it do**
`commentary`	`String`	Commentary for the over.
`commentaryActions`	`Array of Objects (Commentary Action)`	The list of commentary actions.
`inningsIndex`	`int`	The innings index.
`index`	`int`	The over index.
`isComplete`	`Bool`	Is this over complete.

Example Json sent:

```
{
     "method": "getOverCommentary",
     "apiVersion": "1.0.0",
}
```

Example JSON received:

```
{
  "apiVersion" : "1.0.0",
  "commentary" : "End of over 4, 33/1",
  "commentaryActions" :
  [
     {
          "actionType" : 1,
          "bowler" :
          {
               "id" : 21,
               "name" : "RS Bopara",
               "serverId" : "1032691",
               "unsure" : false
          },
          "commentary" : "[bowler] to [batsman], SIX!",
          "facingBatsman" :
          {
               "id" : 1,
               "name" : "CH Gayle",
               "serverId" : "284104",
               "unsure" : false
          },
          "historicalStats" :
          {
               "score" : 39,
               "wicketCount" : 1
          },
          "id" : 19,
          "legalDeliveryIndex" : 0,
          "overIndex" : 3
     }
  ],
  "index" : 3,
  "inningsIndex" : 0,
  "isComplete" : false,
  "methodCaller" : "getOverCommentary"
}
```

Get Match Score View

Method name: `getMatchScoreView`

Output Arguments	Type	What does it do
`awayTeamBasic`	`Object (Team Basic)`	Away team.
`inningsIndex`	`int`	Innings index in play.
`homeTeamBasic`	`Object (Team Basic)`	Home team.
`inningsScoreStats`	`Array of Objects (Innings Batting Score Stats)`	Innings score stats.

isHomeTeamBatting	Boolean	Is home team batting.
matchState	Enum (Match State)	The match state.
matchEndCategory	Emum (Match End Category)	Able to check if there is a winner at the end of a match.

Example Json sent:

```
{
      "method": "getMatchScoreView",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "apiVersion":"1.0.0",
   "awayTeamBasic":{
      "clubServerId":"51785",
      "id":2,
      "name":"Sunrisers Hyderabad, SH",
      "serverId":"51806"
   },
   "homeTeamBasic":{
      "clubServerId":"51777",
      "id":1,
      "name":"Royal Challengers Bangalore, RCB",
      "serverId":"51803"
   },
   "inningsIndex":0,
   "inningsScoreStats":[
      {
         "battingTeam":{
            "clubServerId":"51777",
            "id":1,
            "name":"Royal Challengers Bangalore, RCB",
            "serverId":"51803"
         },
         "endState":0,
         "index":0,
         "scoreStats":{
            "legalDeliveries":4,
            "overIndex":3,
            "score":45,
            "wicketCount":1
         }
      }
   ],
   "isHomeTeamBatting":true,
   "matchEndCategory":-1,
   "matchState":2,
   "methodCaller":"getMatchScoreView"
}
```

Get Batting Bowling View

Method name: `getBattingBowlingView`

Output Arguments	Type	What does it do
bowlerStats	Object (Bowler Stats)	Bowler stats.
facingBatsmanStats	Object (Batsman Stats)	Facing batsman stats.
nonFacingBatsmanStats	Object (Batsman Stats)	Non-facing batsman stats.
nonActiveBowlerStats	Object (Batsman Stats)	Opposite bowler stats.

Example Json sent:

```
{
     "method": "getBattingBowlingView",
     "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
  "apiVersion" : "1.0.0",
  "bowlerStats" :
  {
     "average" : 0,
     "bowler" :
     {
          "id" : 21,
         "name" : "RS Bopara",
         "serverId" : "1032691",
         "unsure" : false
   },
   "legalDeliveriesBowled" : 4,
   "legalDeliveriesBowledCount" : 10,
   "legalDeliveriesBowledIncompleteOversCount" : 0,
   "dotBallsCount" : 1,
   "maidens" : 0,
   "noBallPoints" : 0,
   "overCount" : 1,
   "rpo" : 15,
   "score" : 25,
   "wicketsCount" : 0,
   "widePoints" : 0
},
"facingBatsmanStats" :
{
   "batter" :
   {
         "id" : 1,
         "name" : "CH Gayle",
         "serverId" : "284104",
         "unsure" : false
   },
   "batterOrderIndex" : 0,
   "deliveriesFaced" : 15,
   "dotBallCount" : 1,
   "foursCount" : 4,
   "isDismissed" : false,
   "overCount" : 3,
   "score" : 41,
   "sixesCount" : 3,
   "strikeRate" : 273.33334350585938,
   "timeInPlay" : 0
},
"methodCaller" : "getBattingBowlingView",
"nonActiveBowlerStats" :
{
   "average" : 20,
   "bowler" :
   {
          "id" : 12,
          "name" : "B Kumar",
          "serverId" : "365001",
          "unsure" : false
     },
     "legalDeliveriesBowled" : 0,
     "legalDeliveriesBowledCount" : 12,
     "legalDeliveriesBowledIncompleteOversCount" : 0,
     "dotBallsCount" : 2,
     "maidens" : 0,
     "noBallPoints" : 0,
     "overCount" : 2,
     "rpo" : 10,
     "score" : 20,
     "wicketsCount" : 1,
     "widePoints" : 0
  },
  "nonFacingBatsmanStats" :
  {
     "batter" :
     {
          "id" : 4,
          "name" : "AB de Villiers",
          "serverId" : "283428",
          "unsure" : false
     },
     "batterOrderIndex" : 2,
     "deliveriesFaced" : 1,
     "dotBallCount" : 0,
     "foursCount" : 0,
     "isDismissed" : false,
     "overCount" : 0,
     "score" : 1,
     "sixesCount" : 0,
     "strikeRate" : 100,
     "timeInPlay" : -51
  }
}
```

Method name: `getStatsToWin`

Output Arguments	**Type**	**What does it do**
`dlsInEffect`	`Boolean`	Is DLS is in effect.
`oversRemainingIncludingBreaks`	`int`	Overs renaming including breaks.
`oversUsedIncludingBreaks`	`int`	Overs used including breaks.
`runsToWin`	`int`	Runs to win.
`targetRunRate`	`int`	Target run rate.
`targetScore`	`int`	Target score.

Example Json sent:

```
{
        "method": "getStatsToWin",
        "apiVersion": "1.1.0"
}
```

Example JSON received:

```
TODO
```

Get Duckworth Lewis Stern

Method name: `getDuckworthLewisStern`

Output Arguments	**Type**	**What does it do**
`inEffect`	`Boolean`	Is DLS in effect.
`oversRemainingIncludingBreaks`	`int`	Overs renaming including breaks.
`oversUsedIncludingBreaks`	`int`	Overs used including breaks.
`parScore`	`int`	Par score - it’s the target score if the game was abandoned at that point.
`projectedScore`	`int`	DLS projected score. *Note: this not accurate and will be removed in future updates.*

`runsToWin`	`int`	Runs to win.
`runsToWinPerOver`	`int`	Runs to win per over.
`runsToWinPerOverAverage`	`int`	Runs to win per over on average.
`showResults`	`Boolean`	Should show the results.
`targetScore`	`int`	Target score.
`validResult`	`Boolean`	Is the result valid.
`winningBy`	`int`	Winning by amount.

Example Json sent:

```
{
     "method": "getDuckworthLewisStern",
     "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "methodCaller": "getDuckworthLewisStern",
   "apiVersion": "1.0.0",
   "inEffect" : false,
   "oversRemainingIncludingBreaks" : 20,
   "oversUsedIncludingBreaks" : 0,
   "parScore" : -1,
   "projectedScore" : 153,
   "runsToWin" : -1,
   "runsToWinPerOver" : -1,
   "runsToWinPerOverAverage" : -1,
   "showResults" : false,
   "targetScore" : -1,
   "validResult" : true,
   "winningBy" : -1
}
```

Get Overs Lost

Method name: `getOversLost`
Get the overs lost in a match.

Example Json sent:

```
{
        "method": "getOversLost",
        "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
    "methodCaller": "getOversLost",
    "apiVersion": "1.0.0",
    "oversLost": 0,
}
```

Get Match State

Method name: `getMatchState`
Get the state of the match.

Output Arguments	Type	What does it do
`matchState`	Enum (Match State)	The current state of the match.
`matchEndCategory`	Enum (Match End Category)	Who won at the end of a match.

Example Json sent:

```
{
        "method": "getMatchState",
        "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
    "methodCaller": "getMatchState",
    "apiVersion": "1.0.0",
    "matchState": 0,
}
```

Get Events Count

Method name: `getEventCount`
Get event count.

Example Json sent:

```
{
      "method": "getEventCount",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "methodCaller": "getEventCount",
   "apiVersion": "1.0.0",
   "eventCount": 0,
}
```

Get Over Index

Method name: `getOverIndex`
Get active over index.

Example Json sent:

```
{
      "method": "getOverIndex",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "methodCaller": "getOverIndex",
   "apiVersion": "1.0.0",
   "overIndex": 0,
}
```

Get Innings Index

Method name: `getInningsIndex`
Get active innings index.

Example Json sent:

```
{
      "method": "getInningsIndex",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "methodCaller": "getInningsIndex",
   "apiVersion": "1.0.0",
   "inningsIndex": 0,
}
```

Get Ticker Tape

Method Name: `getTickerTape`
This matches what the application shows. This could be updated over time. Do not parse this text to get data out of it.

Example Json sent:

```
{
     "method": "getTickerTape",
     "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
   "methodCaller": "getTickerTape",
   "apiVersion": "1.0.0",
   "tickerTapeText": "            CricHQ projected score: 277
Batting Team's Win Probability: 31%           Partnership: 12
Lead by 45 off 98 balls            Target run rate: 2.8            ",
}
```

Get Batting Team Analysis

Method Name: `getBattingTeamAnalysis`

Output Arguments	Type	What does it do
`deliveryCount`	`int`	Deliveries bowled.
`dotBallsCount`	`int`	Dot balls count.
`dotBallsPercent`	`int`	Dot balls percentage.
`extraPoints`	`int`	Extra score.
`highestBattingPartnershipScore`	`int`	Highest batting partnership score.

`highestBattingScore`	`int`	Highest batting score.
`maidens`	`int`	Maden count.
`maidenPercent`	`int`	Maiden percent.
`oversCount`	`int`	overs.
`score`	`int`	Score.
`runsFromBoundaries`	`int`	Runs from boundaries.
`runsFromBoundariesPercent`	`int`	Runs from boundaries percent (4’s, 6’s).
`secondHighestBattingPartnershipScore`	`int`	Second highest batting partnership score.
`secondHighestBattingScore`	`int`	Second highest batting score.
`wicketCount`	`int`	Wickets count.
`wideAndNoBallsCount`	`int`	Wide and no balls count.
`wideAndNoBallsPercent`	`int`	Wide and no balls percent.

Example Json sent:

```
{
      "method": "getBattingTeamAnalysis",
      "apiVersion": "1.0.0"
}
```

Example JSON received:

```
{
  "apiVersion" : "1.0.0",
  "deliveryCount" : 22,
  "dotBallsCount" : 3,
  "dotBallsPercent" : 13.636363983154297,
  "extraPoints" : 0,
  "highestBattingPartnershipScore" : 33,
  "highestBattingScore" : 41,
  "maidenPercent" : 0,
  "maidens" : 0,
  "methodCaller" : "getBattingTeamAnalysis",
  "oversCount" : 3,
  "runsFromBoundaries" : 34,
  "runsFromBoundariesPercent" : 75.555557250976562,
  "score" : 45,
  "secondHighestBattingPartnershipScore" : 12,
  "secondHighestBattingScore" : 3,
  "wicketCount" : 1,
  "wideAndNoBallsCount" : 0,
  "wideAndNoBallsPercent" : 0
}
```

Get Event Types Played

Method Name: `getEventTypesPlayed`

Output Arguments	Type	What does it do
eventTypes	Array of Enums (Event Type)	An ordered list of event types.

```
{
  "apiVersion" : "1.0.0",
  "eventTypes" :
  [0, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 22, 22, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 3, 6, 19, 19, 20, 21, 7, 22, 8, 8, 8, 8, 8, 8, 13,
7, 20, 8, 8, 8, 8, 8, 8, 13, 7, 8, 8, 8, 8, 8, 10, 13, 7, 19, 8, 8,
8, 8
  ],
  "methodCaller" : "getEventTypesPlayed"
}
```

Get Last 10 Events

Method Name: `getLast10Events`

Output Arguments	Type	What does it do
events	Array of Events (Event)	An ordered list of the last 10 events.

```
{
   "apiVersion":"1.1.0",
   "events":[
      {
         "eventType":8,
         "guid":"2136140f-becd-462d-998b-aca7935c3515",
         "timestamp":1475697827
      },
      {
         "eventType":8,
         "guid":"e81999c4-d82d-4b06-a23e-701850aaf690",
         "timestamp":1475697827
      },
      {
         "eventType":8,
         "guid":"efa1ace2-537f-49d7-ab80-5aca765fa631",
         "timestamp":1475697827
      },
      {
         "eventType":8,
         "guid":"c201be66-2d90-4218-a962-c30a82565844",
         "timestamp":1475697827
      },
      {
         "eventType":19,
         "guid":"a2d1761a-7f05-497c-b179-a643a4385529",
         "timestamp":1475697827
      },
      {
         "eventType":7,
         "guid":"1bb4dfad-09e2-49f4-92a4-707908531c96",
         "timestamp":1475697827
      },
      {
         "eventType":13,
         "guid":"77cb3324-aa3c-40ad-9eda-cd7bf39ebb37",
         "timestamp":1475697827
      },
      {
         "eventType":10,
         "guid":"ad422873-797a-42a1-ade7-cd38e54eec0b",
         "timestamp":1475697827
      },
      {
         "eventType":8,
         "guid":"15ad026e-d5a8-49e7-94af-28694249bb0d",
         "timestamp":1475697827
```

```
        },
        {
            "eventType":8,
            "guid":"41f6b388-445a-4e87-978b-c2ddbf5966d3",
            "timestamp":1475697827
        }
    ],
    "methodCaller":"getLast10Events"
}
```

Get match type

Method Name: `getMatchType`

Output Arguments	Type	What does it do
`wicketTypes`	`Array of Enums (Wicket Type)`	The wicket types that this match type supports.
`allowBatAgain`	`Boolean`	Allow bat again after a wicket.
`alwaysRebowlNoBalls`	`Boolean`	Always rebowl no balls.
`alwaysRebowlWides`	`Boolean`	Always rebowl no wides.
`capDeliveries`	`Boolean`	Cap deliveries.
`inning2Overs`	`Int`	Innings 1 and 2 overs amount.
`inning4Overs`	`Int`	Innings 3 and 4 overs amount.
`inningsCount`	`Int`	Innings in this match type.
`lastManStands`	`Boolean`	Last man stands.
`legalDeliveriesPerOver`	`Int`	Legal deliveries per over.
`matchName`	`String`	Match name.
`maxDeliveriesPerOver`	`Int`	Max deliveries per over.

`maxOversBatting`	`Int`	Max overs when batting.
`maxOversBowling`	`Int`	Max overs when bowling.
`maxRunsBatting`	`Int`	Max runs when batting.
`noBallValue`	`Int`	No ball points value.
`playersPerTeam`	`Int`	Players per team.
`rebowlLastBallNoBall`	`Boolean`	Rebowl last ball, no ball.
`rebowlLastBallWides`	`Boolean`	Rebowl last ball, wides.
`rebowlLastOverNoBall`	`Boolean`	Rebowl last over, no ball.
`rebowlLastOverWides`	`Boolean`	Rebowl last over, wides.
`unlimitedLastOver`	`Boolean`	Unlimited last over.
`wicketIsDismissal`	`Boolean`	Wicket is dismissal.
`wicketValue`	`Int`	Wicket points value.
`wideValue`	`Int`	Wide points value.

```
{
   "wicketTypes":[
      1, 2, 3, 4, 5, 6, 7, 8, 9, 10
   ],
   "allowBatAgain":false,
   "alwaysRebowlNoBalls":true,
   "alwaysRebowlWides":true,
   "apiVersion":"1.1.0",
   "capDeliveries":false,
   "inning2Overs":20,
   "inning4Overs":-1,
   "inningsCount":2,
   "lastManStands":false,
   "legalDeliveriesPerOver":6,
   "matchName":"Twenty20",
   "maxDeliveriesPerOver":10,
   "maxOversBatting":-1,
   "maxOversBowling":4,
   "maxRunsBatting":-1,
   "methodCaller":"getMatchType",
   "noBallValue":1,
   "playersPerTeam":11,
   "rebowlLastBallNoBall":false,
   "rebowlLastBallWides":false,
   "rebowlLastOverNoBall":false,
   "rebowlLastOverWides":false,
   "unlimitedLastOver":false,
   "wicketIsDismissal":true,
   "wicketValue":0,
   "wideValue":1
}
```

Enum List

Enum name	Info
Match State	The state that the match is in.
Action Type	The type of action.
Match End Category	Who won or lost at the end of a match.
Wicket Type	The type of wicket.

Enums

Match State

Enum Value	Info
2	In progress.
3	Match finished.
5	Match in setup.

6	Match paused.
10	Waiting for innings.
11	Waiting for over.
12	Waiting for the facing batsman.
13	Waiting for the non-facing batsman.
14	Waiting for the bowler.
15	Match finished.

Match End Category

Enum Value	Info
-1	The match is not in an end state.
1	The home team won.
2	The home team lost.
3	Tied.
4	Draw.
5	Abandoned.

Action Type

Enum Value	Info
1	Ball.
2	Extra.
3	Wicket.
4	Penalty.
5	Retire.
6	Injured.

7	Umpire review.
8	Team review.
9	Break.
10	Delay.

Delay type

Enum Value	Info
1	Ground.
2	Weather.
3	Light.
4	Injury.
5	Other.

Break type

Enum Value	Info
1	Drinks.
2	Lunch.
3	Tea.
7	Close of play.
9	Other.
10	Forced. *Deprecated (treat it as “other”).*

Boundary Type

Enum Value	Info

0	None.
1	Four.
2	Six.

Wicket Type

Enum Value	Info
0	Not out.
1	Bowled.
2	Caught.
3	Leg Before wicket.
4	Stumped.
5	Run out.
6	Hit wicket.
7	Handled the ball.
8	Hit the ball twice.
9	Timeout.
10	Obstructing the field.

Extra Type

Enum Value	Info
0	No extra.
1	No ball.
2	Wide.
3	Bye.
4	Leg bye.

Penalty Type

Enum Value	Info
0	None.
1	Fielding.
2	Batting.

Innings End State

Enum Value	Info
0	Not ended.
3	All out.
4	Finished overs.
5	Declared.
6	Closed.
7	Forfeit.
8	Match end.
9	Abandon.
10	DLS delays.

Wicket Type

Enum Value	Info
0	Not out
1	Bowled
2	Caught
3	Leg before wicket

4	Stumped
5	Run out
6	Hit wicket
7	Handled the ball
8	Hit the ball twice
9	Time out
10	Obstructing field

JSON Objects List

Method name	**Info**
Team Basic	Basic information about a team.
Player Basic	Basic information about a player.
Innings Stats	Stats for an innings.
Innings Batting Score Stats	Batting score stats for an innings.
Score Stats	Score stats.
Historical Stats	A stats snapshot in time.
Batsman Stats	Batsman stats.
Bowler Stats	Bowler stats.
Commentary Action	Commentary of an action.
Innings Scorecard	Stats to build a scorecard for an innings.
Action	An over action in the game.
Event	An event that happened in the game

JSON Objects

Many of the API calls return various JSON objects. This section will list them.

Team Basic

This is a basic team object.

Output Arguments	Type	What does it do
`id`	`String`	Team id.
`serverId`	`String`	Team server id.
`clubServerId`	`String`	Club server id.
`name`	`String`	Team name.

Player Basic

This is a basic player object.

Output Arguments	Type	What does it do
`id`	`String`	Player id.
`serverId`	`String`	Player server id.
`unsure`	`Boolean`	Is the player an unsure player.
`name`	`String`	Player name.

Innings Stats

Output Arguments	Type	What does it do
`actionIndex`	`Int`	Current action index. E.g what ball number.
`deliveries`	`Int`	Deliveries.
`boundary6Count`	`Int`	Boundary six count.
`byesScore`	`Int`	Byes score.
`byesCount`	`Int`	Byes count.
`dotBallsCount`	`Int`	Dot balls count.
`dotBallsPercent`	`Int`	Dot balls percentage.

extraScore	Int	Extra score.
legByesScore	Int	Leg bye score.
legByesCount	Int	Leg by count.
legalDeliveries	Int	Legal Deliveries.
maidenCount	Int	Maden count.
maidenPercent	Int	Maiden percentage.
noBallsScore	Int	No balls score.
noBallsCount	Int	No balls count.
overs	Int	Overs.
partnershipScore	Int	Partnership score.
partnershipDeliveries	Int	Partnership deliveries.
penaltiesBattingPoints	Int	Penalties batting points.
penaltiesBowlingPoints	Int	Penalties bowling points.
rpo	float	Runs per over.
score	Int	Score.
runsFromBoundaries	Int	Runs from boundaries.
runsFromBoundariesPercent	float	The percentage of runs that were boundaries.
wicketCount	Int	Wicket count.
wideAndNoBallsCount	Int	Wide and no ball count.
wideAndNoBallsPercent	Int	Wide and no ball percentage.
widesScore	Int	Wide score.
widesCount	Int	Wides count.

Innings batting score stats

Output Arguments	Type	What does it do
battingTeamBasic	Object (Team Basic)	Batting team.
inningsEndState	Enum (Innings End State)	End state of the innings.
inningsIndex	Int	Innings index.
scoreStats	Object (Score stats)	Score stats.

Score stats

Output Arguments	Type	What does it do
legalDeliveries	Int	Legal deliveries.
overs	Int	Overs.
score	Int	Score.
wicketCount	Int	Wicket count.

Historical Stats

Used to store a snapshot of the runs and wickets at that point of time.

Output Arguments	Type	What does it do
score	Int	Score.
wicketCount	Int	Wicket count.

Batsman Stats

Output Arguments	Type	What does it do
deliveriesFaced	Int	The deliveries count that go towards the batter.
batter	Object (Player basic)	Batter.

`batterOrderIndex`	`Int`	Order that the batter batted.
`dotBallCount`	`Int`	Dot ball count.
`boundaryFourCount`	`Int`	Count of four boundaries.
`isDismissed`	`Boolean`	Is dismissed.
`overCount`	`Int`	Overs count.
`score`	`Int`	Score.
`boundarySixCount`	`Int`	Count of six boundaries.
`strikeRate`	`Int`	Strike rate.
`timeInPlay`	`Long`	Time in play.
`outAction`	`Object (Action)`	The action that the player got out in.

Bowler stats

Output Arguments	**Type**	**What does it do**
`legalDeliveriesBowledCount`	`Int`	Total deliveries bowled.
`legalDeliveriesBowled`	`Int`	Deliveries bowled for this over.
`legalDeliveriesBowledIncompleteOversCount`	`Int`	Deliveries bowled that were incomplete overs.
`average`	`Int`	Average.
`bowler`	`Object (Player Basic)`	The bowler.
`dotBallsCount`	`Int`	Dot ball count.
`economy`	`Int`	Economy.
`maidens`	`Int`	Maidens.
`noBallPoints`	`Int`	No ball points.

overCount	Int	Over count.
score	Int	Score.
wicketsCount	Int	Wicket count.
widePoints	Int	Wide points.
rpo	Float	Runs per over.

Commentary Action

Output Arguments	Type	What does it do
id	Int	
actionType	Enum (Action Type)	
bowler	Object (Player Basic)	
commentary	String	
facingBatsman	Object (Player Basic)	
historicalStats	Object (Historical Stats)	
legalDeliveryIndex	Int	
overIndex	Int	

Innings Scorecard

Output Arguments	Type	What does it do
batterListStats	Array of Objects (Batsman Stats)	
bowlerListStats	Array of Objects (Bowler Stats)	
notes	Array of Objects (Notes)	
powerPlays	Array of Objects (Power Play)	

`stats`	`Object (Innings Stats)`	

Action

Output Arguments	Type	What does it do
`id`	`Int`	The ID of the action.
`type`	`Enum (Action Type)`	Action type.
`bowler`	`Object (Player Basic)`	Bowler.
`facingBatsman`	`Object (Player Basic)`	Facing batsman
`inningsIndex`	`Int`	Innings index.
`keeper`	`Object (Player Basic)`	Keeper.
`nonFacingBatsman`	`Boolean`	Non facing batsman
`overIndex`	`Int`	The over index.
`legalDeliveryIndex`	`Int`	The legal delivery index.
`ignoreStats`	`Boolean`	Should this action be ignored (grey it out in the UI). You would ignore an action after a successful umpire/team review.
`legalDeliveryDisplay`	`Int`	Use when displaying the legal delivery number to the user. Do not use this in calculations.

Ball Action

Extends from Action

`boundaryType`	`Enum (Boundary Type)`	

hasPitchMaps	Boolean	
hasWagonWheels	Boolean	
pitchMapBackSquare	Int	
pitchMapBadBowl	Boolean	
pitchMapPitchSquare	Int	
runs	Int	
wagonWheelCutProportion	Float	
wagonWheelEllipseProportion	Float	
wagonWheelHeightProportion	Float	
wagonWheelHitAngleRadians	Float	
toKeeper	Boolean	

Extra Action

Extends from Ball Action

Output Arguments	**Type**	**What does it do**
extraPoints	Int	Extra points.
extraType	Enum (Extra Type)	The type of Extra.

Wicket Action

Extends from Extra Action

Output Arguments	**Type**	**What does it do**
fielders	Array of Objects (Player Basic)	Fielders that helped got the player out. The first index player being the main contributor.
wicketType	Enum (Wicket Type)	The type of wicket.

`outPlayer`	Object (Player Basic)	The batter that got out.

Retire Action

Extends from Action

Output Arguments	Type	What does it do
`retiredBatsman`	Object (Player Basic)	The batter that retired.

Penalty Action

Extends from Action

Output Arguments	Type	What does it do
`penaltyType`	Enum (Penalty Type)	The type of penalty it was.
`penaltyPoints`	Int	The amount of penalty points.

Event

Output Arguments	Type	What does it do
`eventType`	Enum (Event Type)	The type of event.
`guid`	String	The unique id for the event.
`timestamp`	Int	Timestamp of when the event happened.
`matchGuid`	String	The unique id for the match.
`clientTimestamp`	Int	Device timestamp when the event was created.
`order`	Int	The event order.

Notification Events

Every important event on CricHQ will send a notification to the connected device. For consistency, the JSON includes `methodCaller` which is set as `lastEvent`.

Event Type

Event Id	Event Name	Details about event
-1	No event type	No match event type was set.
0	Match type set	The match type has been selected.
1	Match team set	A match team has been selected.
2	Match team player	A match player.
3	Start match	The match has been started.
6	Start innings	New innings has started.
7	Start over	A new over has started.
8	Ball	A normal ball event (dot ball, 4, 14, etc) has been played.
9	Extra ball	An extra ball event (wide, no ball, etc) has been played.
10	Wicket	A wicket ball event has been played.
11	Penalty	A penalty event has been played.
12	Retire	A retired player event has been played.
13	End over	End of the over.
14	End innings	End of the innings.
16	End match	End of the match.
17	Swap batsman	Swap the batsman in play.
19	New batsman	A new batsman.
20	New bowler	A new bowler.
21	Wicket Keeper	A new wicket keeper.
22	New Captain	A new captain.
25	Undo	Undo last event.

26	Audit remove	Removed an event.
28	Audit add	Add a new event anywhere in the match.
29	Audit Modify	Modified an event.
30	Add note	Added a note.
31	Remove note	Remove a note
32	Add pause note	Added a pause note.
33	Add power play	Power play has started
34	Remove power play	Removed power play.
35	Replace player	Replace player with another player.
40	Resume match	Resumes the match.
41	Audit retire modify	Modified a retired event.
42	Outfield speed	The outfield speed. High-performance event.
43	Bounce level	The bounce level. High-performance event.
44	Turn level	The turn level. High-performance event.
45	Update player	A player update. E.g a name change.
46	Field aggression	Field aggression. High-performance event.
47	Boundary rope regression	Boundary rope regression. High-performance event.
48	Weather	The weather. High-performance event.
49	Player of the match	The player of the match. High-performance event.
50	Set toss	Toss was set.
51	Super over	Not implemented yet.
52	Set run in	Bowler running in. High-performance event.
53	Set danger	Ball danger. High-performance event.
54	Set batting order	Batting order.
55	Umpire review	Umpire review. High-performance event.

56	Team Review	Team Review. High-performance event.
57	Injured player	A player was injured. High-performance event.
58	Uninjured player	A player is no longer injured. High-performance event.
59	Review outcome	The outcome of the review. High-performance event.

Note: New events can be added at any time. Make sure you support ignoring of unsupported new events.

Example JSON

If a new match was just started. A “start match” event would send with the API version.

```
{
      "eventType": 3,
      "methodCaller": "lastEvent",
      "apiVersion": "1.0.0"
}
```

If someone pressed the undo button. It will remove the last event.

```
{
      "eventType": 25,
      "methodCaller": "lastEvent",
      "apiVersion": "1.0.0"
}
```

Errors

Error name	**What is the error**
unsupportedVersion	Unsupported API version.
unsupportedMethod	Unsupported method.
unsupportedMethodParameter	Unsupported method parameter.
deprecatedVersion	The version used is deprecated.
missingParameter	The method parameter is missing.

Unsupported Version

If an API version was sent and not supported an error message would send.

```
{
        "methodCaller": "MethodCaller",
        "error": "unsupportedVersion",
        "errorVersion": "0.9",
        "apiVersion": "1.0.0"
}
```

Unsupported methods

If a method was sent and not supported an error message would send.

```
{
        "methodCaller": "MethodCaller",
        "error": "unsupportedMethod",
        "apiVersion": "1.0.0"
}
```

Unsupported Method Parameter

If a method parameter was sent and not supported an error message would send.

```
{
        "methodCaller": "MethodCaller",
        "error": "unsupportedMethodParameter",
        "errorParameter": "ParameterName",
        "apiVersion": "1.0.0"
}
```

Deprecated Version

The version used is deprecated and will be removed in a future version.

```
{
        "methodCaller": "MethodCaller",
        "error": "deprecatedVersion",
        "errorVersion": "0.9",
        "apiVersion": "1.0.0"
}
```

Missing Method Parameter

If a method parameter was missing, this error message would send.

```
{
      "methodCaller": "MethodCaller",
      "error": "missingParameter",
      "errorParameter": "ParameterName",
      "apiVersion": "1.0.0"
}
```

No Data

If a method is called but unable to return data (end/start of match) it will return this error.

```
{
      "methodCaller": "MethodCaller",
      "error": "noData",
      "apiVersion": "1.0.0"
}
```

Broken Json

If broken Json or no Json was sent, this error message would send.

```
{
      "brokenJson": "-r03w-ti 9dxk bs",
      "error": "brokenJson",
      "apiVersion": "1.0.0"
}
```