Click 2 Call

Scope of Work

Contents

[1. Introduction - 2 -](#_Toc337128610)

[2. Target Audience - 2 -](#_Toc337128611)

[3. Product Overview - 2 -](#_Toc337128612)

[4. High-level architecture diagram - 3 -](#_Toc337128613)

[5. Service features - 3 -](#_Toc337128614)

[6. Web based O&M interface - 5 -](#_Toc337128615)

[7. Customer care interface - 5 -](#_Toc337128616)

[8. Customer (Company) Admin interface - 5 -](#_Toc337128617)

[9. Reporting/ Statistics - 5 -](#_Toc337128618)

[10. Monitoring - 6 -](#_Toc337128619)

[11. Logs - 6 -](#_Toc337128620)

[ANNEX I – LOG REQUIREMENT - 7 -](#_Toc337128621)

[Multi location log processing architecture - 7 -](#_Toc337128622)

[IVR Common Log Format - 7 -](#_Toc337128623)

[1. Introduction - 7 -](#_Toc337128624)

[2. Log Fields description - 7 -](#_Toc337128625)

[3. Mandatory Fields of the log file. - 8 -](#_Toc337128626)

[4. Optional fields - 14 -](#_Toc337128627)

[5. Log File Format. - 14 -](#_Toc337128628)

[6. Log File Naming - 14 -](#_Toc337128629)

[7. Log File Rotation - 15 -](#_Toc337128630)

[8. Number of Log files per Application. - 15 -](#_Toc337128631)

[ANNEX II –Call Out Manager - 16 -](#_Toc337128632)

[Annex III – Java Methods 3](#_Toc337128633)

# Introduction

The purpose of this Document is to provide a detailed Scope of Work (SOW), regarding deployment for Click2Call Application for DIALOG.

# Target Audience

The Document is targeting the Following Group of Audience:

* DIALOG VAS Team/Management Team
* DIALOG Developer community

# Product Overview

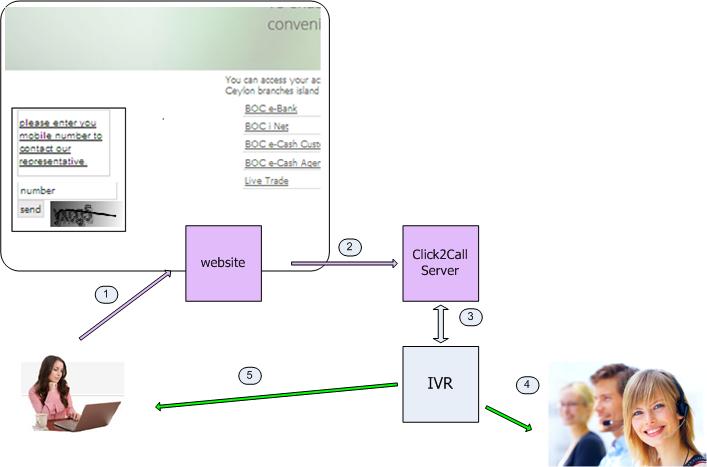
Click to call service enabled customers to contact agent’s through web interface.

Case Study

|  |  |
| --- | --- |
| Priority | Description |
| Phase 1 | As the IT Manager/owner of Enterprise/SME/Startup company want to increase number of sales calls from users visiting company website. Click to call will provide means for website users to initiate free call directly from the website. Charged for number of calls received from web users |
| Phase 1 | As the owner of startup company want all the sales calls initiated from website to get routed to sales representative |
| Phase 1 | As the IT Manager of SME I see running a call center is expensive for us. So I want my operator to run the call center to handle sales call on behalf of us and handle all the sales calls generated from website visitors. |
| Phase 1 | As the IT Manger of enterprise I want all the sales call initiated from company website visitors to get routed to our call center. |
| Phase 2 | As the owner of startup company I would like call charges to be deducted from prepaid call bucket assigned to my company, that I can top up time to time. I need to get alerted, when the balance of my call bucket is going low. |

Target market is Enterprise/ LMB.

# High-level architecture diagram



# Service features

|  |  |
| --- | --- |
| Features | Description |
| Provisioning | * Service activated through Account Manager by TAC via CCRM. * Items involved   Agent Configurations   * + Agent ID   + Agent MSISDN(only DAP)   + Service ID   + Charging   Company Profile   * + Name   + Pilot number for charging   + Registration No.   + Number of Agents   + Black List   + Business Hours   + Authentication model     - Captcha     - SMS Random PIN   + Charging model     - Rental     - Call charge     - Rental + Call charge |
| User Verification | There are two methods for user verification   1. Captcha method   Web site visitor is prompted to enter a captcha to verify call request is actually originated by a Human   1. SMS Random PIN method   This option adds additional layer of verification. Once the user enters captcha and the phone number (Mobile /CDMA Only) to receive the call, system sends out randomly generated PIN number to the phone via SMS. User needs to enter the PIN number before the call request is send to IVR servers  Enterprise is given the option of selecting verification method they wants to use. |
| API | Customer will use text box in iframe to input number and press the button to generate call |
| Black list | * Black listed customers, should be able to add to profile * **Black list users, should be able to add to profile** |
| Charging | * Only offline charging * Charging starts after agent answer the call * Call bucket * Different charging for different Agents/Company * Rental + Call Charges |
| Security | * SSL * OS/SW hardening * Implement IT security policies and pass vulnerability test is a must |
| Notifications | * When reach 90% of utilization SMS notification will be sent to company contact person |
| Call Hunting | * Call hunting is done through IN * Please refer Annex III for API |

# Web based O&M interface

Following features are available in the GUI.

* View profile settings.
* Modify
  + Company Name
  + Number
  + Contact person Name
  + Contact person MSISDN.
  + Configure Agent ID’s
  + Configure Agent Numbers
  + Secret key
* View available charging amount

# Customer care interface

Following features are available in the GUI.

* View profile settings.
* Modify
  + Company Name
  + Number
  + Contact person Name
  + Contact person MSISDN.
  + Configure Agent ID’s
  + Configure Agent Numbers
* View available charging amount

# Customer (Company) Admin interface

Following features should be available from the GUI

* Create/View/modify agents
* Add/Remove numbers to company blacklist
* View company reports/logs

# Reporting/ Statistics

Following information is available.

* Number of profiles/ profile details (Language selection, etc.)
* Total OBD calls per Company
* Total MOU per Company
* MOU per Agent
* Total Revenue (Can be extracted from CG /MIS)

# Monitoring

Application is integrated with Dialog central monitoring system(NRPE or SNMP) to monitor major system/service alarms. Following services/ resources are available for monitoring.

Sample monitoring parameters

* ESME client application.
* Core platform.
* HTTP Service.
* SMPP connectivity.
* CPU usage.
* Memory usage.
* Hard Disk utilization.
* Application overload
* Spoofing for abnormal requests

# Logs

Application logs are generated as per VAS common log format. Refer Annex I for log requirement.

# ANNEX I – LOG REQUIREMENT

## Multi location log processing architecture

**Introduction**

Dialog has several IVR platforms which support VXML 2.0 & any future IVR platform should be compatible with VXML 2.0. By using the VXML 2.0 IVR applications Dialog can archive IVR application/platform redundancy. All the IVR applications should adhere to the common log format

1. Logs will be created in the same server in which the IVR application is in running in.
2. Uploading logs from application servers to IVR dB at 00:05 AM every night will be done using a cron job. Therefore, manual log transfer is not required.
3. This uploading **should be same as** CSV (comma separated) data file for dB uploading.

I.e. uses [MySQL “LOAD DATA”](http://dev.mysql.com/doc/refman/5.0/en/load-data.html)

1. Daily log rotation is also done by this uploading script.
2. Any application that is adhering to the common log format should be able to use same log parser to upload log files to the data base.
3. It should be possible to check statistics from any location using the web based log viewer to query DB.
4. With this architecture by only configuring MySQL command in the common Log interface, log parsing application can be used to get statistic’s from any IVR application which support common log format.

## IVR Common Log Format

1. Introduction

This document is intended to provide a common log format for all the IVR services.

1. Log Fields description

This log file format does not contain a field directive line. Therefore, user should provide the details of the field definitions. The fields should be defined in the order that the fields occur in the log record.

For each field, should provide the following:

#### Field Type

There are several pre-defined field types. The parser will ignore any field with the FID\_IGNORE field type.

Should provide the Field Variable type and Size

i.e.

CHAR(50)

INT( )

DOUBLE( , )

FLOAT()

DECIMAL( , )

DATE()

#### Begin Delimiter

Begin Delimiter is the delimiter that identifies the beginning of the field. The delimiter is not a part of the field.

I.e. comma “,”

If the field begins right after the end delimiter of the previous field, the value for the Begin Delimiter must be null. You cannot use the end delimiter of the previous field as the begin delimiter.

#### End Delimiter

End Delimiter The delimiter that identifies the end of the field. The delimiter is not a part of the field.   
I.e. comma “,’

If the field is the rest of the line, the value of the End Delimiter must be null.

#### Comment Line Token

This token is used to identify the comment line. For an IVR log format, it can be"#" or any string that you want. The only condition is that it should be different from the first field beginning delimiter.

#### NULL Token

This token is used for a field with a null value. For a IVR log format it is "-"

1. Mandatory Fields of the log file.

#### Date

Use standard symbols to define the date format. For a custom log format it is "yyyy-MM-dd "   
where:

|  |  |  |  |
| --- | --- | --- | --- |
| Symbol | Meaning | Format | Example |
| y | year | (Number) | 1996 |
| M | month in year | (Number) | 07 |
| d | day in month | (Number) | 10 |

i.e.

2009-09-16

#### Time

Use standard symbols to define the time format. For a log format it is "HH:mm:ss "   
where:

|  |  |  |  |
| --- | --- | --- | --- |
| Symbol | Meaning | Format | Example |
| H | hour in day (0~23) | (Number) | 0 |
| m | minute in hour | (Number) | 30 |
| s | second in minute | (Number) | 55 |

i.e.

14:03:00

#### Host

Need to include host IP address ([RFC2460](http://tools.ietf.org/html/rfc2460)) which this application is running in.

i.e.

10.48.248.167

#### serviceRefID

This field is used for the service identification.

Called number can be used for this.

I.e. for News IVR this filled is 556

#### CLI

This field should include the calling party MSISDN in [standard format](http://en.wikipedia.org/wiki/MSISDN#MSISDN_Format)

#### CallReferenceID

This field can be used to uniquely identify a particular call.

CallReferenceID is a unique number which may generated by CLI+mmddyy+hh+mm+ss+ms. Or else can use the session ID/reference assigned by the IVR platform.

#### Agent

This field can be used to identify the software version or web server details.

i.e.

tomcat-5.5.20

#### Platform

Which IVR platform processed this call.

I.e. HP, WaveNET, OnMobile, PRBT etc.

#### sessionID

Session ID assigned by platform.

#### Action

This field is used to log different actions of the IVR.

Subscriber & System can make pre defined actions.

i.e.

Login – Each entering to application

DTMF-This is the main actions which indicate the DTMF input by the user.

Traverse- Indicates menu or content traversal without DTMF input.

Charging – Charging request

End Call– End of call

#### Charging

This part covers the possible log fields which should be included in charging request.

#### Action

Should be equal to “Charging”.

#### Operation

i.e. creditCheck

#### CG\_Domain

i.e GSM, PHONE

#### RequestTime

Should be according to the Date/Time format 3.1, 3.2

#### ResponseTime

Should show the response time in milliseconds

#### TransactionKey

CG Transaction key for this transaction

#### User type

Postpaid or Prepaid

#### Amount

Amount debit / credit

#### ReasonCode

Reason code for this application

#### CGResponse

CG response code for this request.

i.e. 0 – successful

5- Service not available

Example log entry for CG transaction

[Date],[Time], [serviceRefID],[host],[Agent],[platform],[CLI], [Action], [CallReferenceID],[sessionID],[ ReasionCode ][ Operation],[ CG\_Domain],[ TransactionKey],[user type],[RequestTime],[ ResponseTime],[ Amount],[ CGResponce],,

2009-09-16, 15:38:20,556, 10.48.248.167, tomcat-5.5.20,OCMP3.2,777856589, Charging,,77785658920090916153820,VXMLsession\_102\_205\_155479, ,postpaid, PHONE, creditCheck,7778565892009091615382045, 15:38:19, 157,10.00,0,,

**Note**

**Log file can consist of manyoptional fields but number of Fields in the log file should be fixed and any empty field should use NULL token.**

**For each service/action, optional field meaning & possible values should be given.**

**Examples**

**Let’s say that a particular log file has 21 fields**

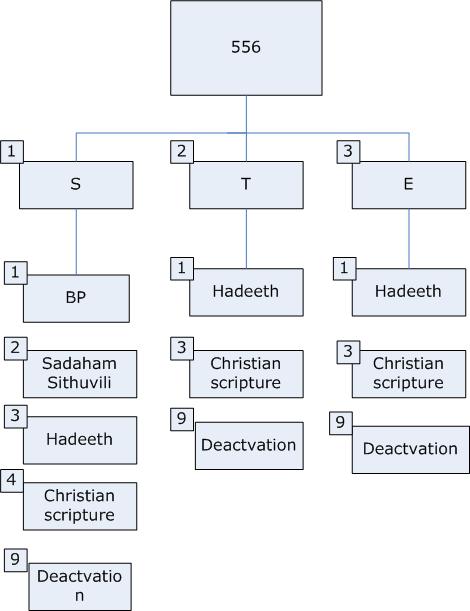
**i.e.**

[Date],[Time], [serviceRefID],[host],[Agent],[platfrom],[CLI], [Action], [CallReferenceID], [sessionID], [ActionDescription],[FIELD12],[FIELD13], [FIELD14], [FIELD15], [FIELD16], [FIELD17], [FIELD18], [FIELD19], [FIELD20], [FIELD21],

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Action** | **Description** | **FIELD 12** | **FIELD 13** | **FIELD 14** | **FIELD 15** | **FIELD 16** | **FIELD 17** | **FIELD 18** | **FIELD 19** | **FIELD 20** | **FIELD 20** |
| Login | Logging to IVR | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| DTMF | LanSellect | Selected language  S/T/E/H | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| DTMF | MenuIn | Menu Language | Parent Menu ID | Current Menu ID | Child Menu | Prompt Listen Duration | NULL | NULL | NULL | NULL | NULL |
| DTMF | MenuOut | Menu Language | Parent Menu ID | Current Menu ID | Next Menu ID | Prompt Listen Duration | NULL | NULL | NULL | NULL | NULL |
| DTMF | CatSelectIn | Selected content language  S/T/E/H | Parent Menu ID | Current Menu ID | CatLevel  In the IVR flow at which level this category located | NULL | NULL | NULL | NULL | NULL | NULL |
| DTMF | CatSelectOut | Selected content language  S/T/E/H |  | Current Menu ID | Cat Level  In the IVR flow at which level this category located | Content Listen duration in seconds | NULL | NULL | NULL | NULL | NULL |
| Traverse/DTMF | ContSelectIn | Selected content language  S/T/E/H | Parent Menu ID | Content Before played | Current Menu ID | Current content | Next Content | NULL | NULL | NULL | NULL |
| Traverse/DTMF | ContSelectOut | Selected content language  S/T/E/H | Parent Menu ID | Content Before played | Current Menu ID | Current content | Next Content | Next Menu ID | Content Listen duration in seconds |  |  |
| DTMF | AddFavCont | Selected content language  S/T/E/H | SongID | CatID | Song Name | Artist | Category | Provider | NULL | NULL | NULL |
| DTMF | AddFavCat | Selected content language  S/T/E/H | CatID | Category | NULL | NULL | NULL | NULL | NULL | NULL | NULL |
| DTMF | Activate | Language | Service ID | Service | CCBS response | Time taken to this request | NULL | NULL | NULL | NULL | NULL |
| DTMF | Dedication | Selected content language  S/T/E/H | Dedicated Mobile No | Song ID | Cat ID | dedication Type  instance/scheduled | Date : Time | Song Name | Artist | Category | Provider |
| DTMF | Charging | Reason Code | Operation | CG Domain | Transaction Key | user type | Request Time | Response Time | Amount | CG Response | NULL |
| DTMF | End | Reason | Total Call Duration | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**Optional fields mean**

#### 556 IVR call flow

****

Sample log entry for logging in to 556 IVR

2009-09-16, 15:38:20,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, Login, 77785658920090916153820150, VXMLsession\_102\_205\_155479,,,,,,,,,,,,

Sample log entry for entering to 556 Languages selection Menu

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, , , 77785658920090916153820150, VXMLsession\_102\_205\_155479, MenuIn,,,lanselectmenu,ServiceSelect, ,,,,,,

Sample log entry for 556 English language selections

2009-09-16, 15:38:30,556, 10.48.248.167, tomcat-5.5.20,HP,777856589, 3, ,77785658920090916153820,VXMLsession\_102\_205\_155479, LanSellect ,E,,,,,,,,,,

Sample log entry for exiting from Language selection menu to English Main menu of 556 IVR

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589,3, 77785658920090916153820150, VXMLsession\_102\_205\_155479, MenuOut, E,, lanselectmenu,EnglishMainMenu,10,,,,,,

Sample log entry for entering 556 English Main Menu

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, 3, 77785658920090916153820150, VXMLsession\_102\_205\_155479, MenuIn,E, lanselectmenu, EnglishMainMenu,,,,,,,,

Sample log entry for leaving 556 English Main Menu & selecting English Christian Scripture reading.

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589,3, 77785658920090916153820150, VXMLsession\_102\_205\_155479, MenuOut ,E,lanselectmenu, EnglishMainMenu,ESCR,8,,,,,,

Sample log entry for entering to 556 English Christian Scripture reading (Note : this is content level)

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, Traverse, 77785658920090916153820150, VXMLsession\_102\_205\_155479, ContSelectIn ,E, EnglishMainMenu,3023.wav, EnglishScriptureMenu,ESCR,3025.wav,EnglishChrisDeact,,,,,,,

Sample log entry for exiting after listening to 556 English Christian scripture content

2009-09-16, 15:38:21,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, Traverse, 77785658920090916153820150, VXMLsession\_102\_205\_155479, ContSelectOut ,E, EnglishMainMenu,3023.wav, EnglishScriptureMenu,ESCR,3025.wav,EnglishChrisDeact,120,,,,,,

Sample log entry for 556 English Christian scripture Service Activation

2009-09-16, 15:38:32,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, 1,77785658920090916153820,VXMLsession\_102\_205\_155479, Activate, ,E,CRS,112 ,OK,2,,,,,,,

Sample log entry for 556 English Christian scripture Deactivation selection.

2009-09-16, 15:38:32,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589, 9,77785658920090916153820,VXMLsession\_102\_205\_155479, Deact, ,E,CRS, ,,,,,,,,,

End Call 556 after caller disconnect

2009-09-16, 15:38:50,556, 10.48.248.167, tomcat-5.5.20, HP, 777856589,,77785658920090916153820,VXMLsession\_102\_205\_155479, End,UserDisconnect ,120,,,,,,,,,,,

**MOD examples**

Sample log entry for 390 adding Sinhala love song “Liyathabara”

(Song ID=557789, Category ID=3003, Category=love,Artist=Athma Liyanage, Provider=Torana) to my favorite

2009-09-16, 15:38:50,390, 10.48.248.167, tomcat-5.5.21,HP,777856589, 3, 77785658920090916153820199,VXMLsession\_102\_205\_155459, S,557789,3003, AddFavCont, Liyathabara, love, Athma Liyanage,Torana,,,

Sample log entry for 390 adding Sinhala Bila Category (Category ID=3003) to my favorite

2009-09-16, 15:38:50,390, 10.48.248.167, tomcat-5.5.21,HP,777856589,4, ,77785658920090916153820199,VXMLsession\_102\_205\_155459,S,,3003, AddFavCat, Bila,,,,,,,,,,

Dedication example

2009-09-16, 15:38:50,390, 10.48.248.167, tomcat-5.5.21,HP,777856589, 5,77785658920090916153820199,VXMLsession\_102\_205\_155459, S,773331158,25346,3006, Dedication ,instance,,Liyathabara, love, Athma Liyanage,Torana

Scheduled dedication

2009-09-16, 15:38:50,390, 10.48.248.167, tomcat-5.5.21,HP,777856589, 3,77785658920090916153820199,VXMLsession\_102\_205\_155459, S,773331158,25346,3006,sheduled,2009-10-05 24:30, Dedication ,iyathabara, love, Athma Liyanage,Torana

Example log entry for CG transaction credit check

2009-09-16, 15:38:20,556, 10.48.248.167, tomcat-5.5.20,OCMP3.2,777856589, Charging,77785658920090916153820,VXMLsession\_102\_205\_155479, postpaid, PHONE, creditCheck,7778565892009091615382045, 15:38:19, 157,10.00,0,,

1. Optional fields

Need to include one of bellow fields in a single log entry, session ID or call reference ID.

1. Log File Format.

Log file format should be in ASCII format and it should be readable from a text editing tool such as WordPad.

## Log File Naming

Format of the log file for a particular date should be

 <YYMMDDXXXX\_NODE\_NAME\_LOG\_NAME>      
  XXXX – File numbering. In case of a large file we can create several log files for same day with different file numbers.

NODE\_NAME – used to identify which IVR platform/Application is used.

I.e. log file name of “MAIN” log for 2009-07-14 & Node WN IVR should be

0907140000\_WN\_MAIN

## Log File Rotation

Log files should be rotated daily by the log uploading application which is described in Multi log processing architecture. Its configuration file should be located in is /conf/ folder of application. It should be possible to configure log file name, location by using the configuration file. Each and every log file should rotate daily at mid night.

Log file name should be according to section 5.

## Number of Log files per Application.

IVR application should use only single log file. That log file should include all the details of the calls & application.

# ANNEX II –Call Out Manager

**1. Introduction**

The Call Out Manager generates outgoing calls by emitting Call Initiation requests to the platform to start the call. The Call Out request, which must be sent to http://*<platformIP>*:5000/callout/request.do, is then forwarded to that node. The Call Out Manager specifies itself as report URL for the Call Initiation requests it emits and is therefore able to keep track of whether the calls fail or succeed. This information is used to enable configurable retry policies. The Call Out Manager uses a scheduling mechanism that permits requests to order outgoing calls in the future. If a call should fail, the Call Out Manager can retry any given number of times. It is possible to make the retry intervals dependent on the reason for failure.

**2. Time in the Call Out Request -The ISO-8601 Standard**

In several places in a Call Out Request times are specified. These are given in the ISO 8601 format. There are several slight variations in the syntax permitted by ISO 8601; a recommended subset is described below.

Basically, the time can be absolute or relative. An absolute time has the form *YYYY-MM-DDTHH:mm:SS*, where *Y* is the year, *M* is the month, *D* is the day, *T* is the capital letter T, *H* is the hour, *m* is the minute and *S* is the second. The part before the *T* can be omitted. A relative time has the form P *YYYY*Y*MM*M*DD*DT*HH*H*mm*M*SS*S, where the non-italic letters represent themselves and the italic letters have the same meaning as above. Any or several of the parts describing year, month and day or hour, minute or second may be omitted.

Example: PT5M means 5 minutes from now. PT2H30M means 2 and half hours from now.

**3. Call Out Request Format**

The following is an example of a Call Out Request that specifies a call to 536722 for the service MyService. The call will be made in 5 minutes, and if it fails the Call Out Manager will retry 5 times. Afterwards, the report will be posted to http://www.mycompany.com/calloutreport\_receiver.jsp.

***Note***

*The "Content-Type" of the Call Out Request as described in the HTTP request must be text/xml.*

|  |
| --- |
| <callout-request version="1.0" xmlns="http://www.dialog.lk/starcall/2011/07/callout-request" xsi:schemaLocation=" http://www.dialog.lk/starcall/2011/07/callout-request callout-request.xsd">  <remote-uri>tel:5367222</remote-uri>  <service id="MyService"/><schedule>  <schedule>  <time>PT5M</time>  <retries>5</retries>  <postpone-on-busy>PT30M</postpone-on-busy>  <postpone-on-noanswer>PT30M</postpone-on-noanswer>  <postpone-on-error>PT30M</postpone-on-error>  <postpone-on-noresource>PT30M</postpone-on-noresource>  </schedule>  <report-url>http://www.mycompany.com/calloutreport\_receiver.jsp</report-url>  </callout-request> |

**Table 3.1.Call Out Request Parameters**

**Parameter Meaning**

|  |  |
| --- | --- |
| Parameter | Meaning |
| remote-uri | This is the telephone number or SIP address to call. Note that the number should start with tel: or sip: to be able to work when the Call Out Manager sends it through Call Initiation. This element may contain a timeout attribute. |
| local-uri | This is the caller’s telephone number or SIP address |
| service | This element should contain an id attribute that specifies the service id. It may contain a url element that contains the URL to the VoiceXML script to run. |
| schedule | This element contains configurations of when the call is to be made. See below |
| fields | A container for additional key-value pairs. These will be propagated to the Call Initiation requests generated by this Call Out request. |
| report-url | This is the URL to which the Call Out reports will be posted. |

**Table 3.2. Call Scheduling Parameters Nestable in the Schedule Element**

|  |  |
| --- | --- |
| Parameter | Meaning |
| time | The time when the call should be made. |
| retries | The number of attempts that should be made before declaring the Call Out a failure. Defaults to 0. |
| postpone-on-busy | The time to wait before retrying when a call failed with the  reason code busy. Defaults to 0. |
| postpone-on-noanswer | The time to wait before retrying when a call failed with the reason code no answer.Defaults to 0. |
| postpone-on-noresource | The time to wait before retrying when a call failed with the reason code no resource Defaults to 0. |
| postpone-on-error | The time to wait before retrying when a call failed with reason code on error. Defaults to 0 |

**4. Call Out Response Format**

The Call Out response is contained within the HTTP response for the Call Out Request. It will contain either an accepted element or a rejected element. The accepted element will have a request-id attribute that will be provided with the Call Out Report to be sent later so client applications can know for which request a given report is sent. The accepted element will contain an absolute ISO 8601 representation of the point in time when the call will be attempted. The rejected element features a reason attribute which will give an error code. The body of the rejected element may contain a more verbose description of the reason for rejection.

* The following is an example of a Call Out response for a successful call scheduling:

|  |
| --- |
| <callout-response version="1.0" xmlns=http://www.dialog.lk/starcall/2011/07/callout-request>  <accepted request-id="http:// www.dialog.lk/callout/show.do?id=1094214804179"2004-09-03T14:38:24.182  </accepted>  </callout-response> |

* The following is an example of a Call Out response for a call scheduling that failed due to a malformed XML request:

|  |
| --- |
| <callout-response xmlns="http://www.dialog.lk/starcall/2011/07/callout-request" version="1.0">  <rejected reason="unknown">Failed retrieving posted document:org.xml.sax.SAXParseException: Attribute name "loc" must be followed by the '=' character.</rejected>  </callout-response> |

**Table 4.1. Values for the reason Attribute of the Rejected Element**

|  |  |
| --- | --- |
| Parameter | Meaning |
| bad-format | The call failed because there was an error in the request. |
| unknown-protocol | The call failed because the protocol of the telephone number or SIP address to call was not |
| no-resource | The call failed because some resource was missing |
| not-authorized | The call failed because some resource was missing. |
| unknown | The call failed because of not further specified error. More information as to the cause can often be found in the body of the rejected element. |

**5. Call Out Report Format**

Below is an example of a Call Out report. Since a Call Out request potentially results in several Call Initiation requests and thus several call attempts the report features a list of these. This report lists three attempts to make a call which implies that the first two failed. The individual elements specify the reason for failure.

|  |
| --- |
| <callout-report xmlns="http://www.dialog.lk/starcall/2011/07/callout-request"  request-id="http:// www.dialog.lk/callout/show.do?id=1094214804179"  status="completed"  version="1.0">  <calls>  <call request-id="[Call ID for first Call Initiation request]"  status="failed"  time="2004-09-15T12:13:14.347"  version="1.0">  <error type="busy"/>  </call>  <call request-id="[Call ID for second Call Initiation request]"  status="failed"  time="2004-09-15T12:23:14.344"  version="1.0">  <error type="no-answer"/>  </call>  <call request-id="[Call ID for third Call Initiation request]"  status="completed"  time="2004-09-15T12:33:14.345"  version="1.0"/>  </calls>  </callout-report> |

# Annex III – Java Methods

|  |  |  |
| --- | --- | --- |
| Package | method | response |
| axiata.dialog.telecommunication.ivr.obd | createCall(  String aPartyNumber,  String bPartyNumber,  String retries,  String schedule,  String applicationID,  String timeout,  URL reportURL,  Boolean debug) | XML status update of the call send to the reportURL |
| axiata.dialog.telecommunication.ivr.charging | checkCredit(  String transKey,  String msisdn,  String reasonCode,  String appID) | CreditCheckObject  Float balance = CreditCheckObject.getBalance(); //remaining balance, could be minus  String accountStatus = CreditCheckObject.getStatus (); //active or deactive  String accountType = CreditCheckObject.getType(); //postpaid/prepaid |
| axiata.dialog.telecommunication.ivr.charging | chargeToBill(  String transKey,  String msisdn,  String reasonCode,  String appID,  Float ammount) | ChargingResponceObject  Float transStatus = ChargingResponceObject.getStatus(); //response from CG  String startTime = ChargingResponceObject..getStartTime(); //transaction start time  String accountType = ChargingResponceObject getEndTime (); //transaction end time | |