

```
1  <?php
2
3
4  /**
5   * Rating Billing and Routing
6   *
7   * Three main classes in this package :
8   * <ol>
9   * <li>e164_call extended by call_ttl.
10  * <li>call_ttl extended by call_routes.
11  * <li>call_routes called by the invite.php (a simple XML-RPC server invoked by a
12  remote client).
13  * </ol>
14  *
15  * Those classes are used by the xml servers (invite.php, invite_stop.php) to calculate
16  and produce:
17  *
18  * <ol>
19  * <li>The Total Time to Live.
20  * <li>The Route(s) a call can/should take
21  * <li>Finally, calculate call costs and profits for Digifonica, Digifonica partners
22  and End users (Partner clients)
23  * </ol>
24  *
25  * @package Digifonica_RBR
26  * @author Fuad A.
27  * @copyright Digifonica
28  *
29  */
30
31
32  require_once("config.php");
33
34
35
36  /**
37   * Number and call type reconstructor.
38   *
39   * e164 is a class specific to the digifonica model, and is used to generate a
40   properly formatted e164 number<br>
41   * according to a caller profile. It goes further to establishing criteria about
42   the callee and its type.<br>
43   * This is a caller-callee reconstructor, no matter where the person resides in the
44   world, s/he can dial local, <br>
45   * national or internationa number the way s/he is used to.<br>
46   * Using a caller profile, we identify if a callee (number being called) is local,
47   national or internal call to <br>
48   * the caller and not to the system.<br>
49   *
50   * Three main classes in this package :
51   * <ol>
```

<b>Voip-Pal Ex. 2014</b> IPR2016-01201
---

```
50      * <li>call_routes called by the invite.php (a simple XML-RPC server invoked by a
remote client).
51      * </ol>
52      *
53      * Those classes are used by the xml servers (invite.php, invite_stop.php) to
calculate and produce:
54      *
55      * <ol>
56      * <li>The Total Time to Live.
57      * <li>The Route(s) a call can/should take
58      * <li>Finally, calculate call costs and profits for Digifonica, Digifonica
partners and End users (Partner clients)
59      * </ol>
60      *
61      * @package Digifonica_RBR
62      * @author Fuad A.
63      * @copyright Digifonica
64      */
65      class e164_call {
66
67
68
69      /**
70      * This is an array containing all possible call errors
71      *
72      * @global Array $errors
73      *
74      */
75      private $errors = array(
76          1=> "1:CALLER PROFILE NOT FOUND",
77          2=> "2:MALFORMED CALLEE,OR NETWORK CALLEE DOES NOT EXIST",
78          3=> "3:WRONG NUMBER DIALED",
79          4=> "4:WRONG INTERNATIONAL NUMBER",
80          5=> "5:WRONG NUMBER LENGTH",
81          6=> "6:WRONG NUMBER LENGTH, LOCAL AREACODE MUST BE DIALED",
82          7=> "7:WRONG NATIONAL NUMBER",
83          8=> "8:CALLED A DID WITH NO MAPPING"
84      );
85
86      /**
87      * This is an array containg all possible call types.
88      *
89      * @var Array
90      */
91      private $call_type = array(
92          1=> "INTERNATIONAL",
93          2=> "NATIONAL",
94          3=> "LOCAL",
95          4=> "NETWORK",
96          5=> "FAIL",
97          6=> "LOCAL,NATIONAL STYLE",
98          7=> "NETWORK, CROSS DOMAIN"
99      );
100
```

```
103     */
104
105     /**
106      * Holds Original caller
107      *
108      * @var String
109      */
110     public $original_caller;
111     /**
112      * Holds original caller domain
113      *
114      * @var String
115      */
116     protected $original_caller_domain;
117     /**
118      * Original caller idd
119      *
120      * @var String
121      */
122     protected $original_caller_idd;
123     /**
124      * Original caller ndd
125      *
126      * @var String
127      */
128     protected $original_caller_ndd;
129     /**
130      * Holds Original caller areacode.
131      *
132      * @var String
133      */
134     protected $original_caller_areacode;
135     /**
136      * Holds Original caller country code.
137      *
138      * @var String
139      */
140     protected $original_caller_countrycode;
141     /**
142      * Holds Original flag if the caller had overlapping numbers
143      *
144      * @var Boolean
145      */
146     protected $original_caller_has_overlapping;
147     /**
148      * Holds Original caller area codes (Overlapping)
149      *
150      * @var Array
151      */
152     protected $original_caller_areacodes=array();
153
154     /**
155      * holds the caller number (Could be a digifonica number or a PSTN number, this is
not set during an incoming PSTN call or a call from another super node.
```

```
158     */
159     public $caller;
160     /**
161      * Holds the caller domain. (not set in the case of incoming call from PSTN or
162      * another supernode)
163      * @var String
164      */
165     protected $caller_domain;
166     /**
167      * Contains callee domain, can be empty when a PSTN call is the callee, if callee
168      * domain is not present, class will atte,pt to guess it.
169      * @var String
170      */
171     protected $callee_domain;
172     /**
173      * The caller Internationa Dialing Digit (IDD) extracted from the caller profile,
174      * to identify an international call. this can have multiple leading Zeroes( treating like
175      * string)
176      * @var String
177      */
178     protected $caller_idd;
179     /**
180      * National Dialing Digit (NDD) for the user, this identify when a user is trying
181      * to make a National call. (can have leading zero, so treating it like a string)
182      * @var String
183      */
184     protected $caller_ndd;
185     /**
186      * the caller area code, this is used to determine if the caller is making a local
187      * call and to find out overlapping area codes if any.
188      * @var integer
189      */
190     protected $caller_areacode;
191     /**
192      * Caller country code, used to determine if a user is making a national call.
193      * @var Integer
194      */
195     protected $caller_countrycode;
196     /**
197      * Flag to identify that a user has overlapping area codes
198      * Example: Vancouver has 604 and 778 and both can call each other for free, so a
199      * call between them is considered local call.
200      * @var Boolean
201      */
202     protected $caller_has_overlapping;
203     /**
204      * This holds all area codes for the user, including overlapping area codes.
```

```
207     */
208     protected $caller_areacodes;
209
210     /**
211      * Flag that the caller has been set.
212      *
213      * @var Boolean
214      */
215     protected $caller_is_set;
216     /**
217      * this is a flag, used to store original callee only once, original callee must be
218      stored in case of multiple call forwards.
219      *
220      * @var Boolean
221      */
222     protected $callee_is_set;
223
224     /**
225      * callee variables
226      */
227     /**
228      * Holds the original callee number
229      *
230      * @var String
231      */
232     protected $original_callee;
233     /**
234      * Original callee domain
235      *
236      * @var String
237      */
238     protected $original_callee_domain;
239     /**
240      * Stores Callee number
241      *
242      * @var string
243      */
244     protected $callee;
245     /**
246      * This is the actual value of the call type, the textual representation is in the
247      array $call_type
248      *
249      * @var integer
250      */
251     protected $callee_type; //look up for $call_type
252     /**
253      * Stores callee data
254      * store as an associative array as follows :
255      * <ol>
256      * <li>ml_id
257      * <li>ml_dialing_code
258      * <li>country_code
259      * <li>ISO_3166_1_alpha_2
```

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.