Real-time Traffic Stats and Costs Monitoring using CGRateS

Dan Christian Bogos
dan.bogos@itsyscom.com

Opensips Summit Chicago, August 2014

Our Background

Over 7 years of experience with architecting server side solutions in VoIP environment

Platform implementations covering both wholesale and retail business categories

Responsibly understanding real-time processing constrains and the seriousness of live system outages

About CGRateS

Modular architecture

Easy to enhance by rewriting specific components
JSON/GOB RPC API

Multi-tenancy support

Built-in task scheduler with support for one time as well as
recurrent actions

Built-in cache system
Data ageing, live counters

Test driven development

About CGRateS (2)

In a Nutshell, cgrates...

⋯ has had 2,082 commits made by 4 contributors
representing 41,835 lines of code

⋯ is mostly written in Go
with an average number of source code comments

⋯ has a young, but established codebase
maintained by a small development team
with increasing Y-O-Y commits

⋯ took an estimated 11 years of effort (COCOMO model)
starting with its first commit in January, 2012
ending with its most recent commit 3 days ago

Languages

| ■ Go | 95% | ■ 7 Other | 5% |

Lines of Code

■ Code   ■ Comments   ■ Blanks
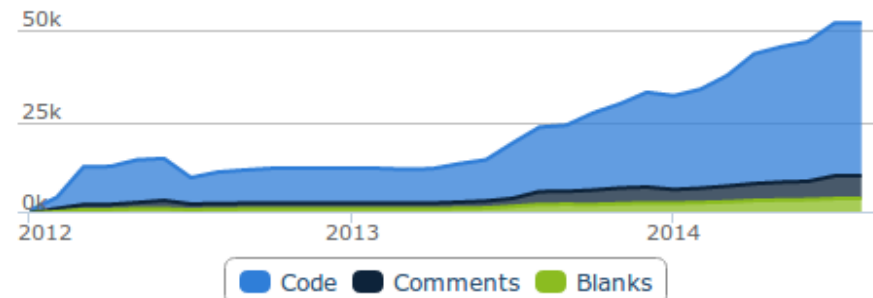
Actively maintained

*stats provided by openhub.net

GRATES
carrier grade realtime charging

About CGRateS (3)


```
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates# date
Mo 4. Aug 14:51:06 CEST 2014
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates# cat /proc/cpuinfo | grep "cpu MHz"
cpu MHz         : 3399.998
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates# cgr-tester -runs=100000
2014/08/04 14:51:36 Runnning 100000 cycles...
2014/08/04 14:51:37 &{*out call cgrates.org 1001  1002 *voice 0.3 [0xc21006d640] true} 99999 <nil>
2014/08/04 14:51:37 memstats before GC: Kbytes = 693 footprint = 5062
2014/08/04 14:51:37 Elapsed: 671817000 resulted: 148850.058870 req/s.
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates# cgr-tester -rater_address="127.0.0.1:2013"
2014/08/04 14:52:01 {*out call cgrates.org 1001  1002 *voice 0.3 [0xc21000fa00] false}
2014/08/04 14:52:01 Elapsed: 805860313 resulted: 12409.098498 req/s.
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates# python /usr/local/src/cgrates/cgrates.git/cgrates/data/tester
(10000, {u'Category': u'call', u'Direction': u'*out', u'TOR': u'*voice', u'Destination': u'1002', u'Account': u'', u'Cost
tId': u'DST_1002', u'TimeEnd': u'2014-04-03T11:13:23.190554134+02:00', u'TimeStart': u'2014-04-03T11:12:23.190554134+02:0
2, 3, 4, 5], u'Years': [], u'StartTime': u'08:00:00', u'EndTime': u''}, u'Rating': {u'ConnectFee': 0.4, u'Rates': [{u'Rat
e': 0.2}, {u'RateIncrement': 1000000000, u'GroupIntervalStart': 60000000000, u'RateUnit': 60000000000, u'Value': 0.1}], u
urationIndex': 60000000000, u'MatchedSubject': u'*out:cgrates.org:call:1001'}], u'Tenant': u'cgrates.org', u'Subject': u'
Elapsed: 1s resulted: 6034 req/s.
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
root@CGRateSDev:/usr/local/src/cgrates/cgrates.git/cgrates#
```

Fast and ... very fast

GRATES
carrier grade realtime charging

Rating

Highly configurable rating
Connect fees, rate units, rate increments, rates grouping, various rounding methods, configurable decimals in costs

Performance oriented
Fully cached

Multiple TypeOfRecord support
(eg: *voice, *data, *sms)

Multiple Category filters for same TOR
(eg: calls, premium_calls, inbound_calls)

CRATES
carrier grade realtime charging

Rating (2)

Multiple rating subjects with fallback
(useful for example with roaming CDRs)

Rating profile scheduling

Rating Aliases

LCR
Monetary
Dynamic/Static
Route filters
Selectable strategy (lowest, highest cost - implemented, QoS - todo)

Git powered History Server

Accounting

Prepaid, Postpaid, Pseudo-prepaid controller

Account Monitors through ActionTriggers
Balance Monitors (min/max)
Counters Monitors (min/max) – eg: Usage per Destination
Synchronous and Asynchronous Actions

Session emulation through DerivedCharging

Concurrent sessions per account

CRATES
carrier grade realtime charging

Accounting (2)

Unlimited Balances per Account
*voice, *data, *sms, *monetary
Balance selection prioritisation through weights

Shared/Group Balances

Balance lifetime controls

Accounts Aliases

CRATES
carrier grade realtime charging

CDRs

Realtime CDR Server

Offline CDR Import (eg: csv format)
Automated via Linux inotify or scheduled

Multiple mediation processes on the same record
Configurable extra fields in cgrates.cfg

CDR Exporter
CSV, Fixed Length Fields, Combined

CRATES
carrier grade realtime charging

CGRates Peripherals

APIer (RPC server)
Tariff plan and Account management
Export commands form internal components (Eg: get_cdrs, export_cdrs, etc)
Partial and full rates/accounts reload without restarts

Console
Interactive and non-interactive
History
Help
Command auto-completion

Loader
CSV Imports

Tester

CDR Stats Server

Standalone component
Internally or remotely accessible
Performance oriented

RawCDR and RatedCDR sources

Multiple Stats Queues
Per server and individually configurable stat queues for same CDR

Highly configurable Stats Queues
QueueLength, TimeWindow, Metrics
CDR Field Filters

Individually configured ActionTriggers
One-time, recurrent triggers
Synchronous & Asynchronous Actions executed

ƆRATES

carrier grade realtime charging

OpenSIPS Integration Tests

Rated call scenario

Account 1001
ASR/ACD/ACC out of tests in multiple stats queues modified
No balance affected

Postpaid call scenario

Account 1002
ASR/ACD/ACC out of tests in multiple stats queues
Balance of 1002 altered

Pseudoprepaid call scenario

Account 1003
ASR/ACD/ACC out of tests in multiple stats queues
Balance of 1003 altered
Dialog timeout occurs if balance exhausted

ƆRATES
carrier grade realtime charging

Where to go from here

Website
http://www.cgrates.org

Documentation
http://cgrates.readthedocs.org

Code + issues tracker
https://github.com/cgrates/cgrates

Support
Google group: CGRateS
IRC Freenode: #cgrates

CGRATES
carrier grade realtime charging

Thank you!

Questions?

CRATES

carrier grade realtime charging