# CÁC THUẬT NGỮ VIẾT TẮT

| Thuật ngữ | Định nghĩa |
| --- | --- |
| HLD | High level design |
| LLD | Low level design |
| VOCS | Viettel Online Charging System |
| OCS | Hệ thống tính cước trực tuyến (Online Charging System) |
| OCU | Phân hệ tính cước (Online Charging Unit) |
| CGW | Thành phần thu thập các yêu cầu tính cước trực tuyến (Charging Gateway) |
| SCP | Hệ thống điều khiển service logic, data, cuộc gọi (Service Control Point) |
| OCP | Online Charging Process |

**THÔNG TIN LIÊN HỆ**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Họ và tên** | **Email** | **Đơn vị** |
| **1** | Nguyễn Trọng Thái | thaint@viettel.com.vn | TT.KTTC-VTNET |
| **2** | Phạm Ngọc Hiệu | hieupn@viettel.com.vn | TT.KTTC-VTNET |

I. GIỚI THIỆU CHUNG

1.1. Mục đích tài liệu

Tài liệu cho hệ thống tính cước thời gian thực vOCS3.0 Lab Viettel PTIT

1.2. Đối tượng sử dụng tài liệu

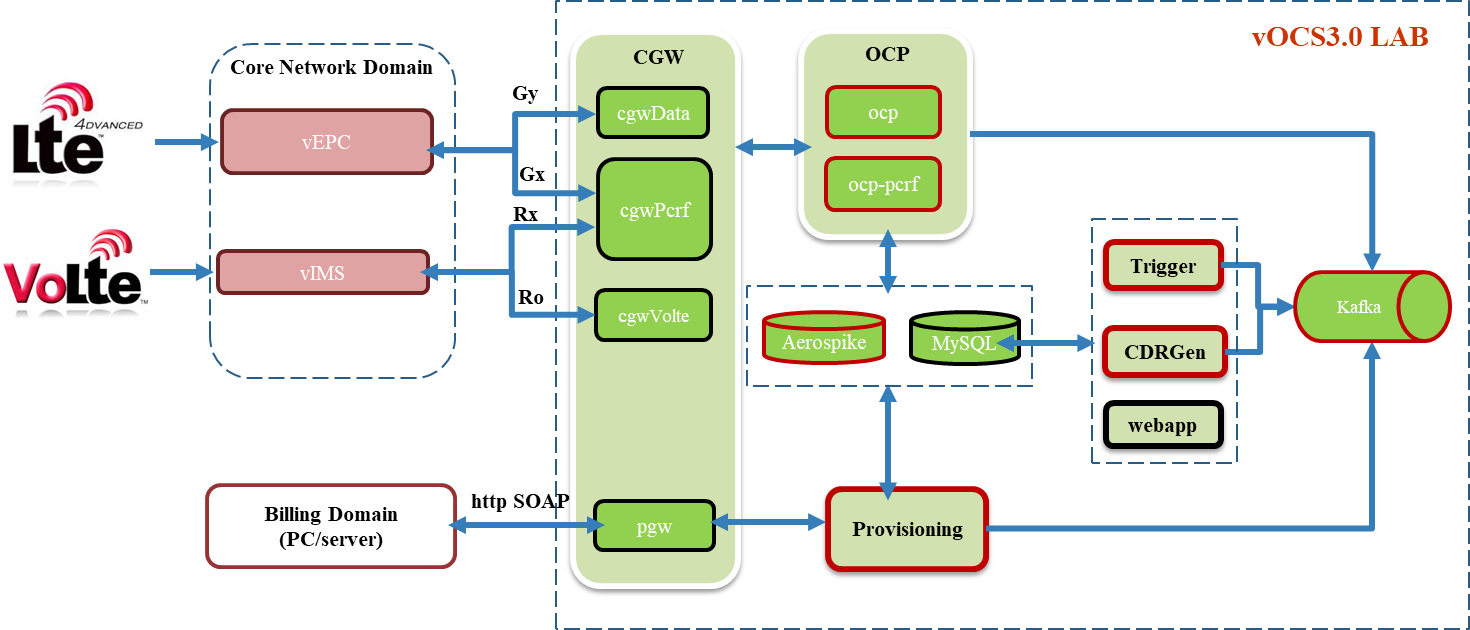
1.3. Lịch sử tài liệu

1.4. Các dịch vụ đáp ứng

|  |  |
| --- | --- |
| **Các dịch vụ trên vOCS3.0 Lab Viettel PTIT** | **Trạng thái** |
| Dịch vụ Data 4G | x |
| Dịch vụ VOLTE | x |

# II. TỔNG QUAN VỀ HỆ THỐNG

2.1. Sơ đồ kết nối logic hệ thống vOCS



2.2. Kiến trúc tổng thể hệ thống vOCS3.0 Lab Viettel PTIT

Hệ thống vOCS 3.0 chia làm 2 phân hệ lớn chính:

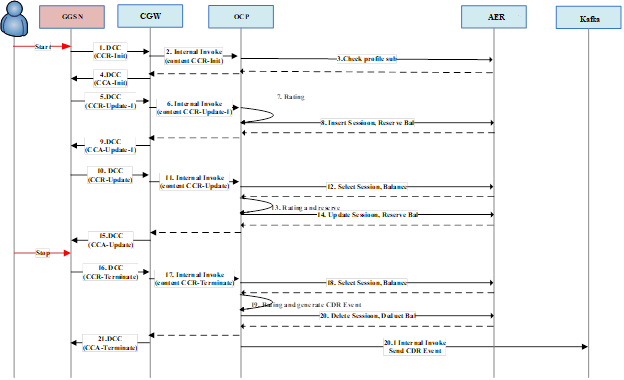
* **OCU (Online Charging Unit):** Phân hệ thực hiện tính cước, cung cấp các cơ chế, giao diện và nghiệp vụ tính cước.
* **OAM (Operation and Administion Management):** Phân hệ hỗ trợ vận hành, thống kê và giám sát toàn bộ hệ thống vOCS.

2.3. Chức năng các phần tử trong hệ thống

* **Gateway (ChargingGW và ProvisioningGW)**: Phân hệ đóng vai trò là cổng giao tiếp tính cước với các hệ thống khác. Hệ thống bao gồm 2 loại gateway là Charging gateway (Đảm nhiệm kết nối với các hệ thống GGSN, MMSC, SMSC, BRAS) và Provisioning gateway (Đảm nhiệm kết nối tới các hệ thống Provisioning, TRAN, VAS). Phân hệ Gateway cũng đóng vai trò điều hướng và phân phối bản tin tới OCP và ProvisioningProcess.
* **OCP** (Online Charging Processor): là thành phần chính của OCU, thực hiện chức năng tính cước thời gian thực cho tất cả các dịch vụ và điều khiển băng thông cho các dịch vụ data (Mobile data, FTTH, ADSL, …).
* **ProvisioningProcess**: Là phân hệ thực hiện xử lý các nghiệp vụ các nghiệp vụ đăng kí, thay đổi, hủy gói cước, tạo nhóm, thêm thành viên, hủy nhóm, nạp tiền, thay đổi tài, truy vấn tài khoản, recurring, ....
* **WebApp** (Web Application Server): Giao diện cấu hình gói cước.
* **Aerospike IMDB**: Là cơ sở dữ liệu lưu trữ thông tin hồ sơ (profile) và tài khoản của khách hàng trên bộ nhớ chính (RAM- Bộ nhớ truy cập nhanh nhằm hỗ trợ tăng hiệu năng xử lý thời gian thực cũng như các nghiệp vụ khác).
* **MySQL**: Là cơ sở dữ liệu vật lý lưu trữ thông tin cấu hình và gói cước.
* **TriggerProcess**: Thực hiện phân loại trigger liên quan đến Notify tới khách hàng và các trigger liên quan đến tạo ra các event tác động tới profile khách hàng như cộng tích lũy, đăng kí gói cước. Đầu ra của phân hệ TriggerProcess là các sự kiện thông báo (SMS\_Notify, USSD\_Notify) để gửi tới phân hệ Notifier hoặc các sự kiện tính cước để gửi tới phân hệ Gateway.
* **CDR**: Bao gồm CDRGen thực hiện chức năng sinh file CDR, CDRPush thực hiện đẩy file CDR sang hệ thống Billing và CDRHolding lưu trữ CDRFile vào cơ sở dữ liệu lịch sử.
* **Tool support**: Công cụ hỗ trợ mô phỏng, khai báo gói cước, logic, trừ tiền, sinh CDR, … sử dụng test thử gói cước trước khi áp dụng trên hệ thống thật.
* **SCU (Service Control Unit)**: Hệ thống điều khiển dịch vụ, cuộc gọi.
* **SCP (Service Control Point)** Là khối điều khiển dịch vụ, quản lý logic của các dịch vụ voice, ussd, ivr. SCP giao tiếp với BEP và CGW để gửi bản tin tính cước cho thuê bao.
* **OAM (Operation and Administion Management)** giám sát toàn bộ các tiến trình, node mạng của hệ thống vOCS3.0 (SCU, OCU, VC):
* **OAM DB:** OAM lưu trữ dữ liệu cấu hình giám sát và dữ liệu giám sát thu thập được từ các client trên Database Mysql

2.4 FLOW DỊCH VỤ vOCS3.0

### **2.4.1 Flow tính cước Data**



Hình 16: Flow dịch vụ Data

Bước 1: Thuê bao nằm trong hệ thống OCS thực hiện truy cập data, GGSN sẽ kiểm tra các thuộc tính tính cước cho dịch vụ này và gửi một bản tin khởi tạo phiên lên OCS.

Bước 2: CGW server nhận được yêu cầu và chuyển tới thành phần nội bộ, sau đó bản tin được gửi tới OCP.

Bước 3: OCP thực hiện giải mã bản tin, lấy thông tin msisdn, sau đó truy vấn IMDB AeroSpike để kiểm tra thông tin thuê bao và gửi bản tin CCA cho CGW.

Bước 4: CGW gửi bản tin CCA tới GGSN để trả lại thông tin xác thực thuê bao cho GGSN.

Bước 5: GGSN gửi bản tin CCR (Update - lần 1) cho CGW để yêu cầu reserve data.

Bước 6: CGW server nhận được yêu cầu và chuyển tới thành phần nội bộ, sau đó bản tin được gửi tới OCP.

Bước 7: OCP phân tích thông qua luồng Pre-Processing để lấy được event tính cước. Sau đó, OCP thực hiện luồng rating để tính toán cước phí cho thuê bao.

Bước 8: OCP thực hiện tạo session truy cập và reserve tài khoản vào trong các bảng của database IMDB AeroSpike, sau đó gửi bản tin phản hồi CCA cho CGW.

Bước 9: Sau khi OCP mã hóa bản tin gửi CGW server, CGW gửi CCA (Update, GSU,Return-Code) tới GGSN.

Bước 10: Khi thuê bao sử dụng hết lượng reserve hoặc reserve hết hiệu lực, GGSN gửi bản bản tin CCR update tiếp theo tới CGW server. Bản tin này chứa thông tin về lưu lượng data thuê bao đã sử dụng sau bản tin CCA (Update – lần 1).

Bước 11: CGW server nhận được yêu cầu và chuyển tới thành phần nội bộ, sau đó bản tin được gửi tới OCP.

Bước 12: OCP lấy thông tin phiên truy cập và thông tin tài khoản thuê bao từ IMDB AeroSpike để chuẩn bị thực hiện luồng rating.

Bước 13: OCP thực hiện luồng rating, dựa vào thông tin lưu lượng data thuê bao sử dụng ở Bước 10 để tính cước cho thuê bao. Sau khi hoàn thành tính cước cho thuê bao, OCP dựa vào thông tin tài khoản hiện tại và cấu hình giá để thực hiện reserve thêm lưu lượng data cho thuê bao.

Bước 14: OCP thực hiện cập nhật lại session và reserve tài khoản trên IMDB AeroSpike, sau đó gửi bản tin CCA tới CGW.

Bước 15: CGW gửi bản tin CCA (Update, GSU, Return-Code) tới GGSN để thông báo lưu lượng data được cấp phát cho thuê bao.

Bước 16: Khi khách hàng dừng sử dụng dịch vụ, GGSN gửi bản tin CCR (Terminate) tới CGW để yêu cầu tính lại cước cho thuê bao.

Bước 17: CGW server nhận được yêu cầu và chuyển tới thành phần nội bộ, sau đó bản tin được gửi tới OCP.

Bước 18: OCP thực hiện lấy thông tin session và thông tin tài khoản của thuê bao từ IMDB AeroSpike để chuẩn bị cho luồng rating.

Bước 19: OCP thực hiện luồng rating để tính cước cho thuê bao và sinh CDR event.

Bước 20: OCP thực hiện trừ tiền tài khoản, xóa session trên IMDB AeroSpike.

Sau đó, OCP gửi CDR Event tới CdrGen để sinh ra file CDR lưu thông tin về chi tiết cước và gửi bản tin CCA tới CGW.

Bước 21: CGW gửi bản tin CCA (Terminate, Return-Code) tới GGSN. Toàn bộ quá trình data kết thúc.

### **2.4.2 Flow luồng VOLTE**

#### 2.4.2.1. Luồng MO



1. Khi thuê bao bắt đầu thực hiện cuộc gọi, hệ thống IMS gửi bản tin CCR-I lên hệ thống OCS. Module tiếp nhận bản tin là CGW.
2. Tiến trình CGW thực hiện chuyển đổi bản tin thành bản tin nội bộ và gửi xuống tiến trình OCP
3. Tiến trình OCP tiếp nhận bản tin, thực hiện khởi tạo session, cấp phát reserve cho cuộc gọi, sau đó gửi bản tin trả về cho CGW. Trong tiến trình OCP, luồng xử lý bản tin Volte được xử lý tương tự như luồng voice.

Trong trường hợp phát media thông báo theo mã lỗi tương ứng, thì OCP đóng thêm Announcement-Instructions (avp-code = 1441, vendor Ericsson). Bảng mapping mã lỗi và announcement number:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Case | Ocs\_code | Ims\_code | Ann\_number | media file |
| Hết tiền đủ điều kiện ứng tiền | 2025 | 4012 | 40001 |  |
| Hết tiền không đủ điều kiện ứng tiền | 2007 | 4012 | 40002 |  |
| Thuê bao thiếu thông tin | 2008 | 4100 | 41001 |  |
| Idle | 2011 | 4100 | 41002 |  |
| Chặn 1 chiều | 2009 | 4222 | 42001 |  |
| Chặn 2 chiều | 2010 | 4222 | 42002 |  |
| Suspend - Chặn do nạp thẻ sai quá số lần quy định | 2012 | 4222 | 42003 |  |
| Lost - Chặn thuê bao do mất máy | 2013 | 4222 | 42004 |  |
| Chặn 1 chiều do khách hàng yêu cầu | 2020 | 4222 | 42005 |  |
| Thuê bao thực hiện cuộc gọi ngoài zona | 2021 |  |  |  |

1. Tiến trình CGW chuyển đổi bản tin Internal thành bản tin Diameter và gửi trả về IMS
2. Thời điểm hết reserve, hệ thống IMS gửi bản tin CCR-U lên hệ thống OCS.
3. Tiến trình CGW thực hiện chuyển đổi bản tin thành bản tin nội bộ và gửi xuống tiến trình OCP
4. Tiến trình OCP tiếp nhận bản tin, thực hiện:
   * update session,
   * tính cước lượng sử dụng mà IMS notify lên trong bản tin CCR-U,
   * cấp phát reserve tiếp theo cho cuộc gọi,
   * gửi bản tin trả về cho CGW.
5. Tiến trình CGW chuyển đổi bản tin Internal thành bản tin Diameter và gửi trả về IMS
6. Thuê bao tiếp tục cuộc gọi. Hệ thống IMS định kì gửi bản tin CCR-U lên hệ thống OCS lên khi lượng reserve được sử dụng hết
7. Thuê bao kết thúc cuộc gọi, hệ thống IMS gửi bản tin CCR-U lên hệ thống OCS.
8. Tiến trình CGW thực hiện chuyển đổi bản tin thành bản tin nội bộ và gửi xuống tiến trình OCP
9. Tiến trình OCP tiếp nhận bản tin, thực hiện:
   * delete session,
   * tính cước lượng sử dụng mà IMS notify lên trong bản tin CCR-T,
   * gửi bản tin trả về cho CGW.
10. Tiến trình CGW chuyển đổi bản tin Internal thành bản tin Diameter và gửi trả về IMS

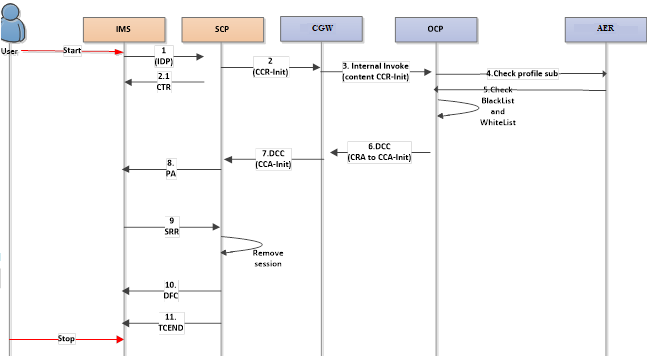
#### 2.4.2.2. Luồng MF

Khi thuê bao A thực hiện cuộc gọi cho thuê bao B, thuê bao B cấu hình forward cuộc gọi đến thuê bao C thì hệ thống IMS sẽ gửi tạo ra 2 phiên tính cước:

* + Phiên 1: thông tin Calling là số A, Called là số B 🡪 OCS tính cước như một cuộc gọi thông thường
  + Phiên 2: thông tin Calling là số A, Called là số C và Redirecting-Party-Address là số B 🡪 là phiên call forward, OCS tính cước cho số B.

Luồng bản tin của từng phiên tương tự như luồng MO.

#### 2.4.2.3. Luồng MT



Mô tả các bước xử lý thiết lập luồng MT LTE

**Bước 1:** Khi khách hàng thực hiện cuộc gọi MT LTE thì IMS gửi bản tin báo hiệu IDP với eventypeBcsm và servicekey khác lên hệ thống ocs

**Bước 2:** Scp khi nhận được bản tin IDP sẽ lưu lại sessionid của bản tin này và tạo bản tin CCR-I gửi sang CGW

**Bước 2.1:** Scp thực hiện gửi đồng thời bản tin CTR về cho IMS

**Bước 3:** CGW nhận biết theo eventtypeBcsm là 12 set thuộc tính Mtflag là 1, rồi gửi bản tin internal msg CGW sang OCP

**Bước 4:** Ocp sẽ thực hiện truy vấn AeroDb để lấy thông tin profile của thuê bao,

**Bước 5:** Db Aerodb trả về phản hồi trong đó sẽ có danh sách các số blacklist, whitelist cho OCP

**Bước 6:** Ocp trả CGW về cho cgw có kèm theo mã lỗi

**Bước 7:** Cgw chuyển đổi bản tin CGW thành CCA-I trả về cho scp

**Bước 8:** Scp nhận được bản tin CCA-I dựa vào mã lỗi, tương ứng với từng mã lỗi khác nhau sẽ thực hiện gửi bản tin PA với id các bài IVR khác nhau ra hệ thống IMS

**Bước 9:** Khi IMS phát xong bài hát thì sẽ thực hiện gửi bản tin SRR về cho SCP. SCP thực hiện xóa sessionid lưu trên SCP ở bước 2

**Bước 10:** SCP đẩy bản tin DFC ra để ngắt luồng media

**Bước 11:** SCP thực hiện đẩy bản tin TCEND ra để giải phóng dialog dưới BEP và để IMS cũng giải phóng dialog

Mô tả các bước xử lý tính cước cuộc gọi MT

Sau khi IMS và SCP thiết lập thành công luồng gọi MT thì IMS sẽ thực hiện đẩy bản tin qua giao diện Ro lên cgwVolte như một luồng MO như phía trên để charge cước thuê bao

# III. Hệ thống vOCS3.0 Lab Viettel PTIT

3.1 Thông tin hệ thống vOCS3.0 Lab Viettel PTIT

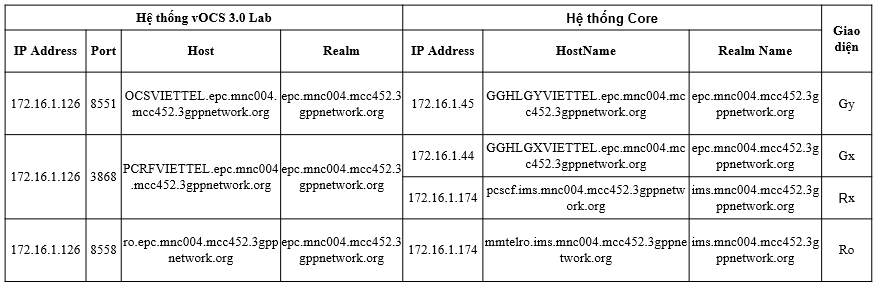
| **NODE** | **IP** | **Tiến trình** | **Đường dẫn** | **user** | **Pass** |
| --- | --- | --- | --- | --- | --- |
| OCS01 | 10.70.66.13 | Pgw | /u01/ocu/pgw/ | vt\_admin | [123456a@](mailto:123456a@) |
| cgwData | /u01/ocu/cgwData |
| cgwVolte | /u01/ocu/cgwVolte |
| cgwPcrf | /u01/ocu/cgwPcrf |
| webapp | /u01/apache-tomcat-8.5.14 |
| oam-agent | /u01/oam/oam-agent |
| Mysql | /u02/mysql |
| OCS02 | 10.70.66.14 | Zookeeper, Kafka | /u02/zkuser | vt\_admin | [123456a@](mailto:123456a@) |
| OCP-Pcrf | /u01/ocu/ocp-pcrf |
| Provisioning | /u01/ocu/provisioning |
| OCP | /u01/ocu/ocp |
| oam-agent | /u01/oam/oam-agent |
| CDRGEN | /u01/ocu/cdrgen |
| Trigger | /u01/ocu/trigger |
| Aerospike | /etc/aerospike |
| Linkoam | [http://10.70.66.4](http://10.70.66.4/) | user/pass:hvbcvt/Hvbcvt@123 |  |  |  |
| linkweb | <http://10.70.66.13:8080/vocs_lab/login.xhtml> | user/pass:admin/Ha2911@@ |  |  |  |
| link webtool | <http://10.70.66.13:8080/WebTool/login.xhtml> | user/pass:vocs/123456a@ |  |  |  |
| linkSoap | 10.70.66.13:8102?wsdl | user/pass:vocs3/vocs3 |  |  |  |

3.2 Các lệnh Linux cơ bản

* + - * 1. pwd command
      * Command**pwd** được dùng để tìm đường dẫn của thư mục hiện tại (folder) mà bạn đang ở trong đó. Command này sẽ trả về đường dẫn hoàn chỉnh (đầy đủ), bắt đầu bằng dấu gạch chéo (/). Ví dụ một đường dẫn hoàn chỉnh là **/home/username**.
      * 2. Command cd
      * Để chuyển hướng trong hệ thống tập tin Linux, bạn có thể sử dụng command **cd**. Nó sẽ cần nhập đường dẫn đầy đủ hoặc tên thư mục bạn muốn chuyển tới.
      * Nếu bạn đang ở trong **/home/username/Documents** và muốn đến **Photos**, thư mục con của **Documents**, chỉ cần gõ **cd** **Photos**.
      * Trường hợp khác là nếu bạn muốn chuyển sang danh mục hoàn toàn mới, như **/home/username/Movies**. Lúc này, bạn phải gõ **cd** theo danh mục đường dẫn hoàn chỉnh như sau:
      * cd /home/username/Movies
      * Có nhiều cách di chuyển nhanh bằng cd như sau:
      * **cd ..** (với 2 chấm) để chuyển lên 1 cấp thư mục trên
      * **cd**để tới thẳng thư mục home
      * **cd-** (với dấu gạch ngang) để chuyển tới thư mục bạn đã ở trước đó
      * Mặt khác, shell của Linux phân biệt chữ hoa chữ thường nên bạn phải gõ tên danh mục thật chính xác.
      * 3. Command ls
      * Command ls được dùng để xem nội dung thư mục. Mặc định là command này sẽ hiển thị danh sách file trong thư mục hiện tại.
      * Nếu bạn muốn xem nội dung thư mục khác, hãy nhập **ls** và sau đó là đường dẫn thư mục. Ví dụ: nhập **ls** **/home/username/Document** để xem nội dung của **Documents**.
      * Có nhiều phiên bản để dùng với lệnh **ls**như sau:
      * **ls -R** liệt kê các file bao gồm cả các thư mục phụ bên trong
      * **ls -a** liệt kê những file ẩn
      * **ls -al**liệt kê tất cả file và thư mục với thông tin chi tiết như phân quyền, kích thước, chủ sở hữu, vân vân.
      * 4. Command cat
      * **cat** là một trong các lệnh cơ bản trong Linux được sử dụng thường xuyên nhất trong Linux. Nó được dùng để xem nội dung file trên output tiêu chuẩn (sdout). Để chạy lệnh này, gõ **cat** theo sau là tên file và phần mở rộng**.**Ví dụ: **cat file.txt.**
      * Có nhiều cách để sử dụng **cat** command linux:
      * **cat > filename**tạo ra file mới
      * **cat filename1 filename2>filename3**nhập 2 files (1 và 2) để lưu kết quả vào file (3)
      * để chuyển một file từ in thường tới in hoa hoặc ngược lại, **cat filename | tr a-z A-Z >output.txt**
      * 5. Command cp
      * Sử dụng command **cp** để sao chép files từ thư mục hiện tại. Chẳng hạn, command **cp scenery.jpg** **/home/username/Pictures** sẽ tạo bản copy của **scenery.jpg** vào danh mục **Pictures**.
      * 6. Command mv
      * Công dụng chính của command **mv** là di chuyển files, dù nó cũng có thể được dùng để đổi tên files.
      * Arguments trong command này tương tự như command cp. Bạn cần nhập **mv**, tên file và điểm đến của thư mục. Ví dụ: **mv file.txt /home/username/Documents**.
      * Để đổi tên files, cú pháp là **mv oldname.ext newname.ext**
      * 7. Command mkdir
      * Command **mkdir** đượ dùng để tạo thư mục mới – giống như **mkdir Music** sẽ tạo thư mục mới gọi là **Music**.
      * Một số cách dùng cộng thêm của lệnh **mkdir**:
      * Để tạo một thư mục mới bên trong thư mục khác, sử dụng lệnh Linux cơ bảnsau: **mkdir Music/Newfile**
      * sử dụng **p**(parents) option để tạo thư mục giữa 2 thư mục đã tồn tại. Ví dụ, **mkdir -p Music/2020/Newfile** sẽ tạo thư mục “2020”
      * 8. Command rmdir
      * Nếu bạn cần xóa thư mục, sử dụng command **rmdir**. Tuy nhiên, rmdir chỉ cho phép bạn xóa các thư mục trống.
      * 9. Command rm
      * Command **rm** được sử dụng để xóa thư mục cùng và nội dung bên trong. Nếu bạn chỉ muốn xóa thư mục – tương tự như lệnh rmdir – sử dụng **rm -r**.
      * **Lưu ý**: Khi dùng các lệnh cơ bản trong Linux bạn cần cẩn thận, đặc biệt là lệnh này. Bạn cần kiểm tra kỹ bạn đang ở thư mục nào. Nó sẽ xóa mọi thứ và không khôi phục được.
      * 10. Command touch
      * Command **touch** cho phép bạn tạo files mới trống thông qua dòng lệnh. Ví dụ: nhập touch **/home/username/Documents/Web.html** để tạo file HTML tiêu đề **Web** trong thư mục **Documents**.
      * 11. Command locate
      * Bạn có thể sử dụng lệnh này để **locate** (định vị) file, giống như lệnh tìm kiếm trong Windows. Hơn nữa, việc sử dụng argument **-i** với lệnh này làm cho nó không còn phân biệt chữ hoa chữ thường, nên bạn có thể tìm file ngay cả khi không nhớ tên chính xác.
      * Để tìm file chứa hai hoặc nhiều từ, hãy sử dụng dấu hoa thị **(\*)**. Ví dụ: command**locate -i school\*note** sẽ tìm tất cả file nào chứa từ “school” và “note”, không phân biệt chữ hoa hay chữ thường.
      * 12. Command find
      * Tương tự như command locate, command**find** cũng tìm files. Sự khác biệt là bạn sử dụng command find để xác định vị trí files trong thư mục nhất định.
      * Ví dụ, command find **/home/ -name notes.txt**sẽ tìm file tên **notes.txt** trong thư mục chính và thư mục con của nó.
      * Một vài biết thể để dùng lệnh **find** là:
      * Để tìm file trong thư mục hiện tại, dùng lệnh **find . -name notes.txt**
      * Để tìm thư mục được dùng, **/ -type d -name notes. txt**
      * 13. Command grep
      * Đây là một trong số các lệnh cơ bản trong Linux hữu ích được dùng hằng ngày. Command **grep** cho phép bạn tìm kiếm tất cả text thông qua tập tin nhất định.
      * Để minh họa, **grep blue notepad.txt** sẽ tìm từ **blue**trong file notepad. Các dòng có chứa từ được tìm sẽ hiển thị đầy đủ.
      * 14. sudo command
      * Command sudo là viết tắt của **“SuperUser Do”**, cho phép bạn thực hiện các tác vụ yêu cầu quyền quản trị hoặc quyền root. Tuy nhiên, không nên sử dụng lệnh này hàng ngày vì dễ xảy ra lỗi nếu làm sai.
      * 15. Command df
      * Command **df** dùng để nhận báo cáo về dung lượng lưu trữ được sử dụng trên hệ thống, hiển thị theo tỷ lệ phần trăm và KBs. Nếu bạn muốn xem báo cáo tính bằng megabyte, hãy nhập **df -m**.
      * 16. Command du
      * Nếu bạn muốn kiểm tra dung lượng của file hoặc của thư mục, command **du** (Disk Usage – Dung lượng lưu trữ) sẽ làm chuyện này. Tuy nhiên, bản tóm tắt về dung lượng lưu trữ sẽ hiển thị block numbers của ổ đĩa thay vì định dạng kích thước thông thường. Nếu bạn muốn xem theo byte, kilobyte và megabyte, hãy thêm argument **-h** vào dòng lệnh.
      * 17. Command head
      * Command **head** được sử dụng để xem dòng đầu tiên của bất kỳ file văn bản nào. Theo mặc định, nó sẽ hiển thị 10 dòng đầu tiên, nhưng bạn có thể thay đổi số này theo ý mình. Ví dụ: nếu bạn chỉ muốn hiển thị 5 dòng đầu tiên, hãy nhập **head -n 5 filename.ext**.
      * 18. Command tail
      * Command này có chức năng tương tự như command head, nhưng thay vì hiển thị dòng đầu tiên, command **tail** sẽ hiển thị 10 dòng cuối cùng của file văn bản. Ví dụ, **tail -n filename.ext.**
      * 19. diff command
      * Viết tắt của difference, command **diff** sẽ so sánh nội dung của 2 files từng dòng một. Sau khi phân tích files này, nó sẽ xuất ra các dòng không khớp nhau. Lập trình viên thường dùng lệnh này khi cần thực hiện một số thay đổi chương trình thay vì viết lại toàn bộ mã nguồn.
      * Hình thức đơn giản nhất của lệnh này là **diff file1.ext file2.ext**
      * 20. Command tar
      * Command**tar** là command được sử dụng rộng rãi nhất để lưu trữ nhiều file vào **tarball** – một định dạng file Linux phổ biến tương tự định dạng zip, nhưng nén file thì tùy.
      * Lệnh này khá phức tạp với danh sách chức năng dài như thêm files mới vào một archive hiện có, liệt kê nội dung của archive, giải nén nội dung từ archive và nhiều chức năng khác. Xem lại [**một số ví dụ thực tế**](https://www.linuxtechi.com/17-tar-command-examples-in-linux/) để biết thêm về các chức năng khác.
      * 21. Command chmod
      * Command**chmod** là một command thiết yếu khác, dùng để thay đổi quyền đọc, ghi và quyền thực thi files và thư mục. Vì lệnh này khá phức tạp nên bạn có thể đọc [**hướng dẫn đầy đủ**](https://www.computerhope.com/unix/uchmod.htm) để thực hiện đúng.
      * 22. Command chown
      * Trong Linux, tất cả files được sở hữu bởi một người dùng cụ thể. Command **chown** cho phép bạn thay đổi hoặc chuyển quyền sở hữu file sang tên người dùng được chỉ định. Chẳng hạn, **chown linuxuser2 file.ext** sẽ biến **linuxuser2** thành chủ sở hữu **file.ext**.
      * 23. Command jobs
      * Command **jobs** sẽ hiển thị tất cả jobs hiện tại và trạng thái jobs. Job cơ bản là một tiến trình được tạo bởi shell.
      * 24. Command kill
      * Nếu có chương trình nào đó không phản hồi, bạn có thể chấm dứt chương trình thủ công bằng cách sử dụng command **kill.** Nó sẽ gửi tín hiệu nhất định đến những ứng dụng đang hoạt động sai và hướng ứng dụng tự chấm dứt.
      * Có tổng cộng [**64 tín hiệu**](https://linoxide.com/linux-how-to/linux-signals-part-1/) có thể sử dụng, nhưng mọi người thường chỉ dùng 2 tín hiệu:
      * **SIGTERM (15)** – yêu cầu chương trình ngừng chạy và chờ một chút để lưu tất cả tiến trình. Nếu bạn không chỉ định tín hiệu khi nhập command kill, tín hiệu này sẽ được sử dụng.
      * **SIGKILL (9)** – buộc các chương trình phải dừng ngay lập tức. Tiến trình nào chưa lưu sẽ bị mất.
      * Ngoài việc biết về tín hiệu, bạn cũng cần biết Số nhận diện quá trình (PID) của chương trình muốn **kill**. Nếu bạn không biết PID, chỉ cần chạy lệnh **ps ux**.
      * Sau khi biết tín hiệu muốn dùng và PID của chương trình, hãy nhập cú pháp sau:
      * **kill [signal option] PID**.
      * 25. Command ping
      * Sử dụng command **ping** để kiểm tra trạng thái kết nối của bạn với server. Ví dụ: bằng cách nhập **ping google.com**, lệnh sẽ kiểm tra xem bạn có thể kết nối với Google hay không và đo thời gian phản hồi.
      * 26. Command wget
      * Một dòng lệnh cực kỳ hữu ích của Linux – bạn có thể tải file từ internet xuống với sự trợ giúp của command **wget**. Để làm được, chỉ cần gõ **wget,** đằng sau là link tải xuống.
      * 27. Command uname
      * Command **uname**, viết tắt của Unix Name, sẽ in thông tin chi tiết về hệ thống Linux của bạn như tên máy, hệ điều hành, kernel, v.v.
      * 28. Command top
      * Là terminal tương đương với Task Manager trong Windows, command **top** sẽ hiển thị danh sách tiến trình đang chạy và lượng CPU mà tiến trình đó sử dụng. Rất có ích khi bạn giám sát dung lượng lưu trữ tài nguyên trên hệ thống, đặc biệt là biết được quá trình nào cần chấm dứt vì tiêu thụ quá nhiều tài nguyên.
      * 29. Command history
      * Khi bạn sử dụng Linux một thời gian nhất định, bạn sẽ nhanh chóng nhận thấy bạn đang chạy hàng trăm lệnh mỗi ngày. Vì vậy việc chạy command **history** đặc biệt hữu ích nếu bạn muốn xem lại những command bạn nhập trước đó.
      * 30. Command man
      * Bạn nhầm lẫn về chức năng của các commands? Đừng lo, bạn có thể học cách sử dụng chúng dễ dàng từ shell của Linux bằng cách dùng command **man**. Ví dụ, nhập **man tail** sẽ hiển thị hướng dẫn thủ công của command **tail**.
      * 31. Command echo
      * Lệnh này được dùng để chuyển dữ liệu vào một file. Ví dụ, nếu bạn muốn thêm text “Hello, my name is John” vào trong file  name.txt, bạn nhập ệnh **echo Hello, my name is John** **>> name.txt**
      * 32. Command zip, unzip
      * Sử dụng lệnh **zip** để nén file thành zip archive và lệnh **unzip** để giải nén file zipped trong zip archive.
      * 33. Command hostname
      * Nếu bạn muốn biết tên của một máy / mạng máy tính bạn chỉ cần gõ **hostname**. Thêm option **-I** vào cuối file để hiển thị địa chỉ IP address của network bạn.
      * 34. Command useradd, userdel
      * Vì Linux là một hệ máy đa người dùng, có nghĩa là nhiều hơn 1 người tương tác với máy cùng lúc. **useradd** được dùng để tạo user mới, còn **passwd** đặt mật khẩu cho một tài khoản user. Để thêm user có tên John, **useradd John** rồi thiết lập password bằng lệnh **passwd 123456789.**
      * Để xóa một user cũng tương tự như thêm user. Để xóa user account, nhập lệnh **userdel UserName**

3.3 Khai báo kết nối trên node CGW

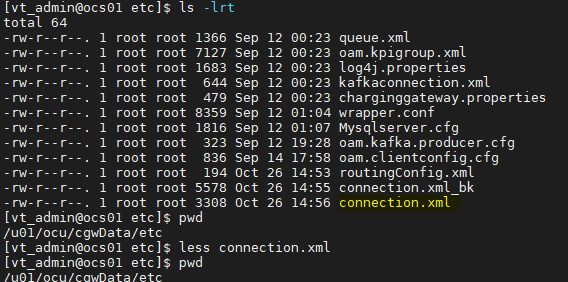
### **3.3.1 Thông tin kết nối giữa vOCS 3.0 với mạng Core**

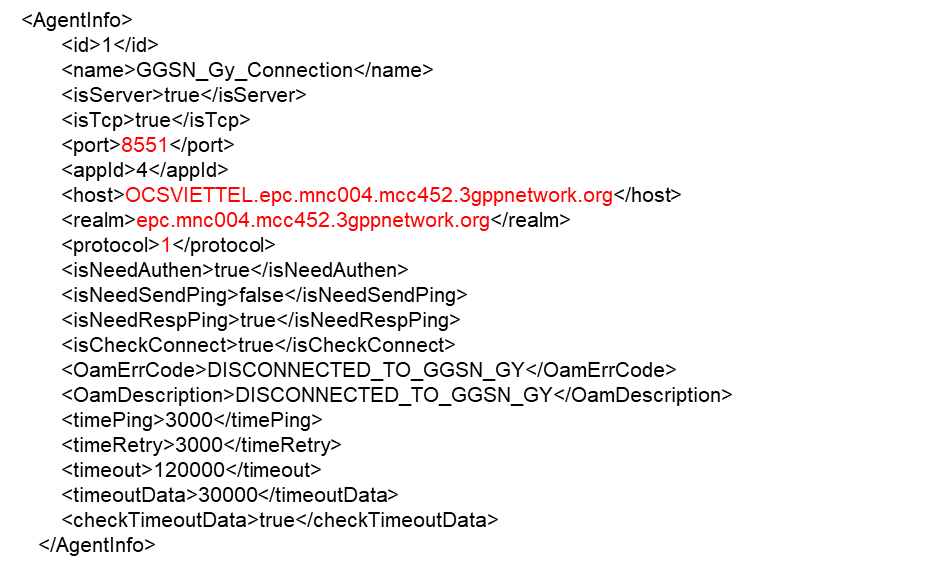


### **3.3.2 Cấu hình khai báo kết nối**

**Ví dụ:** kết nối giao diện Gy

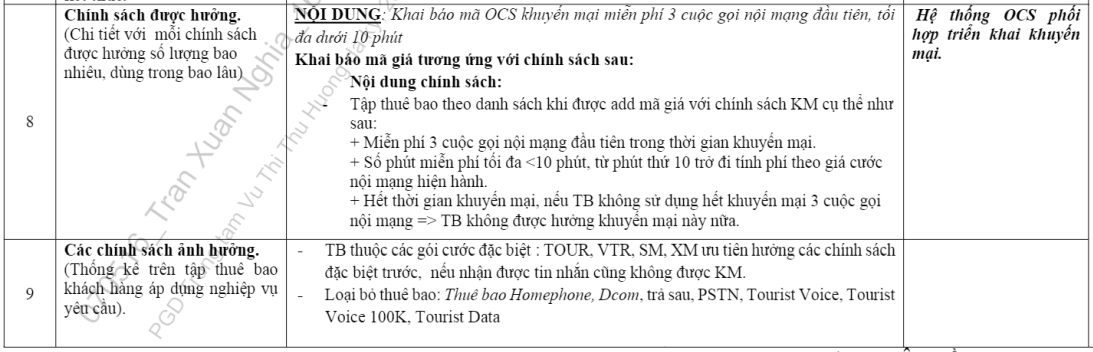
/u01/ocu/cgwData/etc/connection.xml



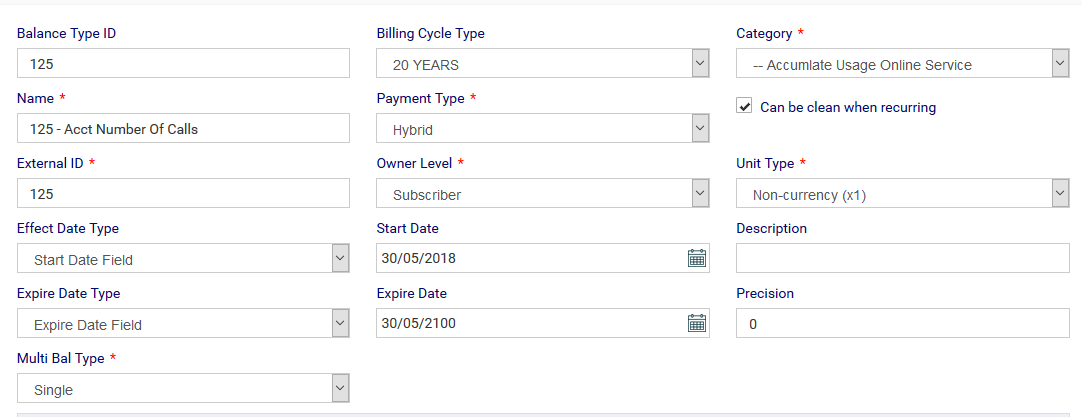


3.4 Cấu hình gói cước

Ví dụ bài toán:



# Tạo tài khoản tích lũy riêng số cuộc gọi cho gói cước này

****

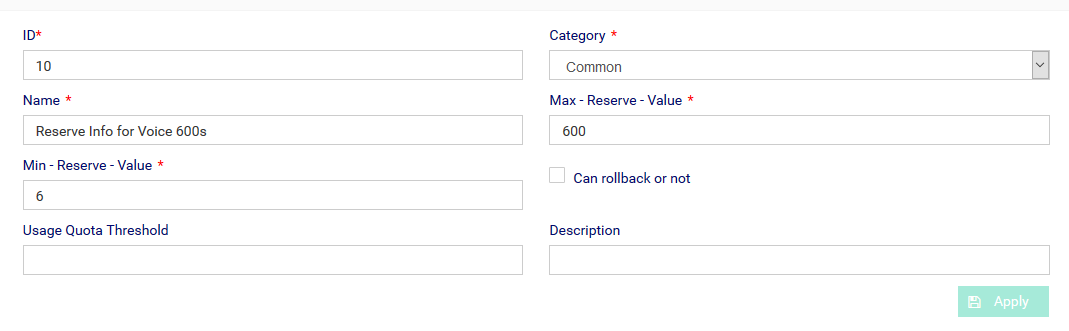
Thực hiện update bảng bal\_type để set bal\_type\_id về giá trị theo mình mong muốn

update bal\_type set bal\_type\_id = 125 where bal\_type\_name = '125- Acct Number Of Call';

# Tạo Reserve Infor để reserve riêng cho gói cước này

Giả sử chính sách cuộc gọi đươc free 10 phút (tức là 600s) thì cấu hình

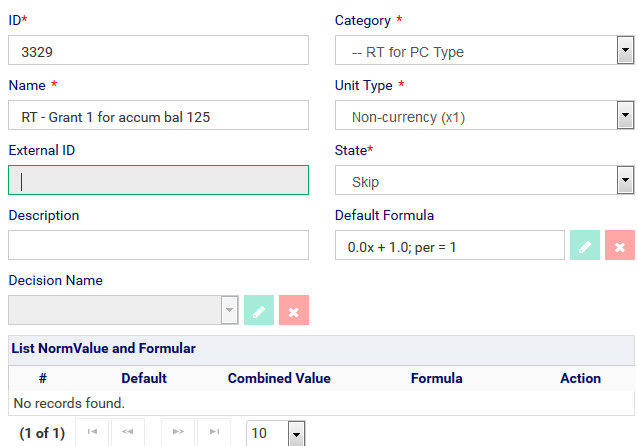
* min reserve = 6
* max\_reserve = 600
* Trên hệ thống đã tồn tại sẵn reserve infor này

****

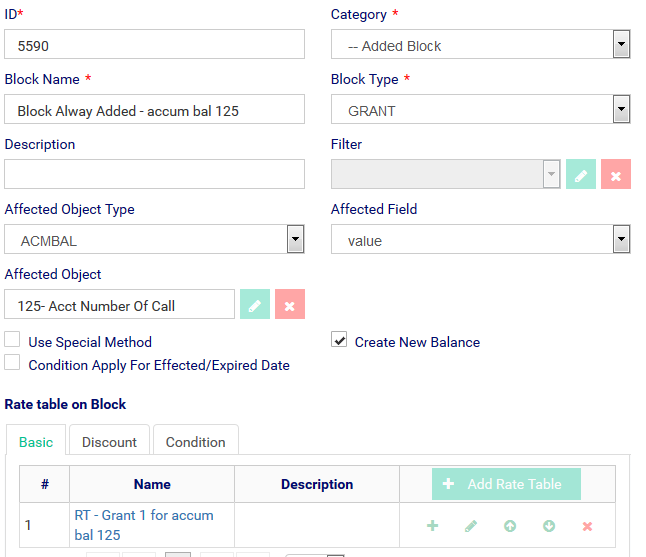
# Khai báo cấu hình tỉnh cước

## **Khai báo block tăng tài khoản accum 125 lên 1**

* **Tạo RT - Grant 1 for accum bal 125**

****

* **Tạo block tăng tài khoản 125**

****

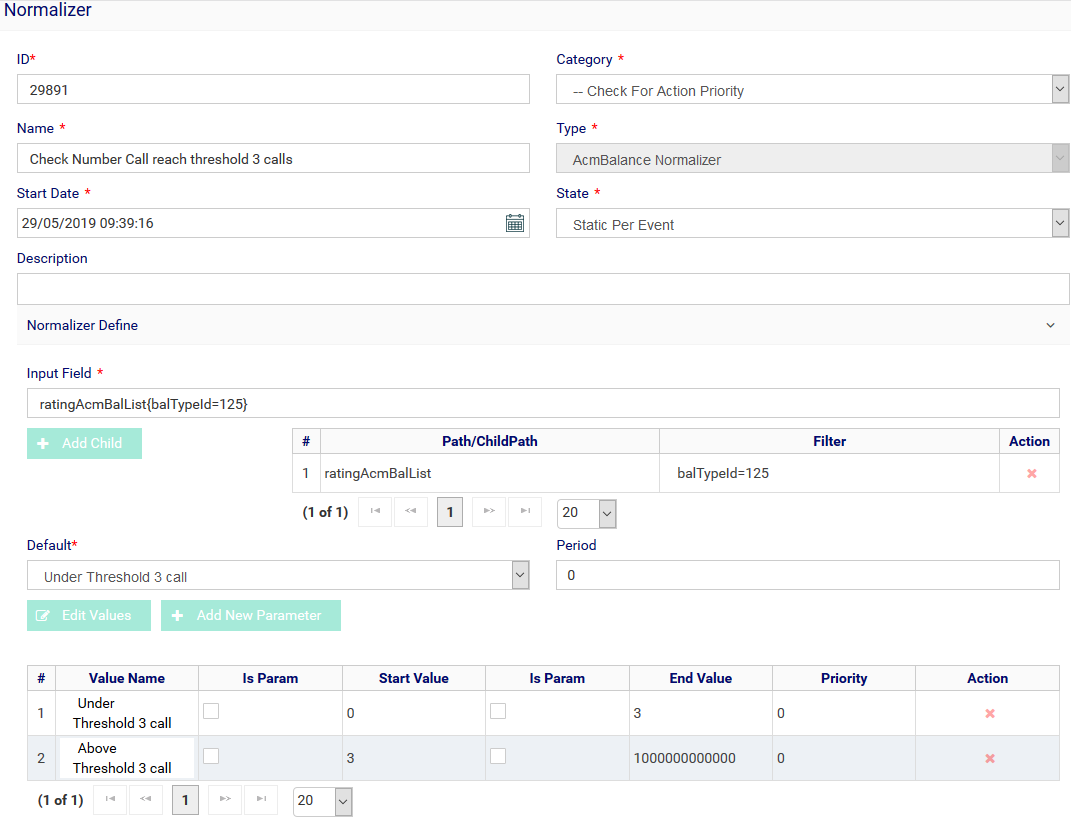
## **Tạo PC Onnet Price Following VTFREE Offer 0/600s BalType 1 for 3 first Call**

Gồm 2 block:

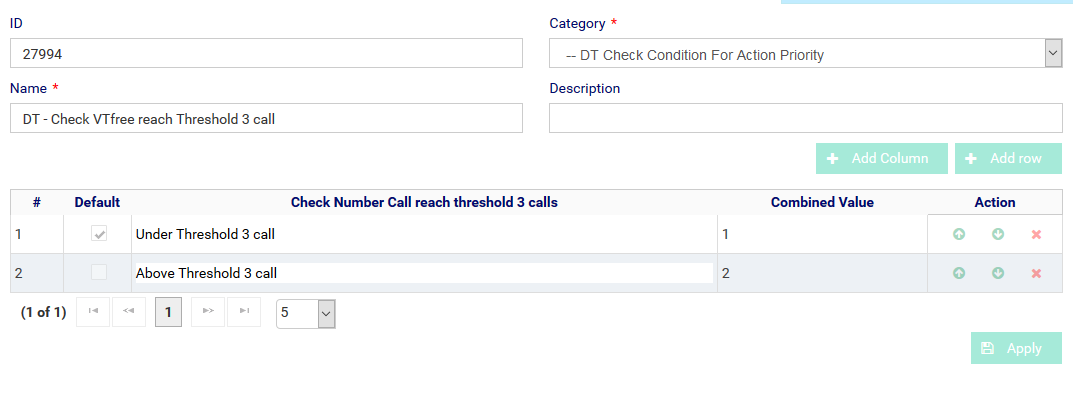
* Block Onnet 0-600 Price 0vnd - Other Exit - BalType1
* Block Alway Added - accum bal 125

## **Tạo PriorityFilter để check nếu số cuộc gọi vượt quá 3 cuộc thì nhảy sang gói cước khác**

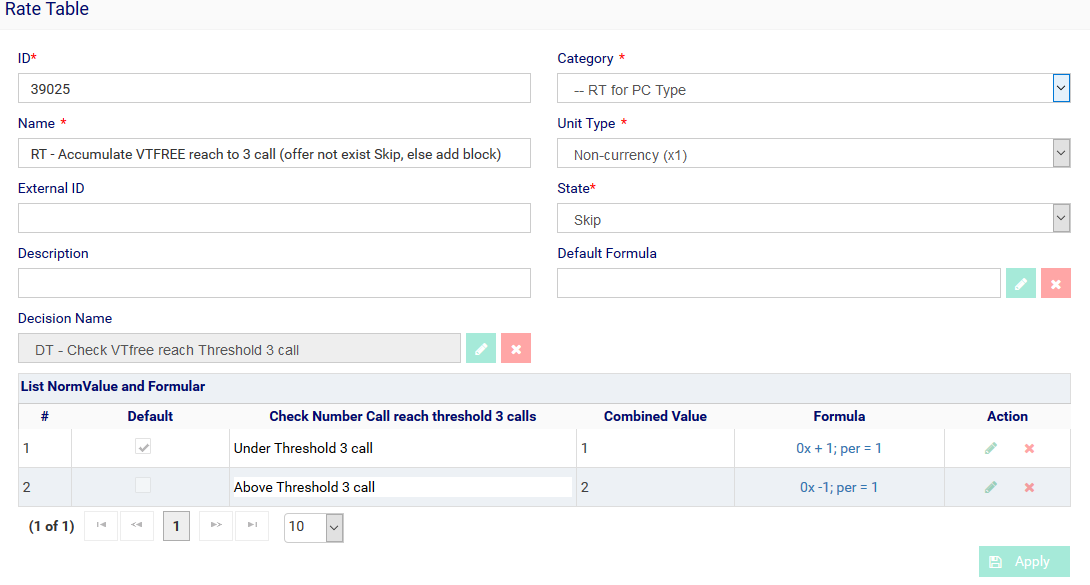
* Tạo Normarlizer Check Number Call reach threshold 3 calls



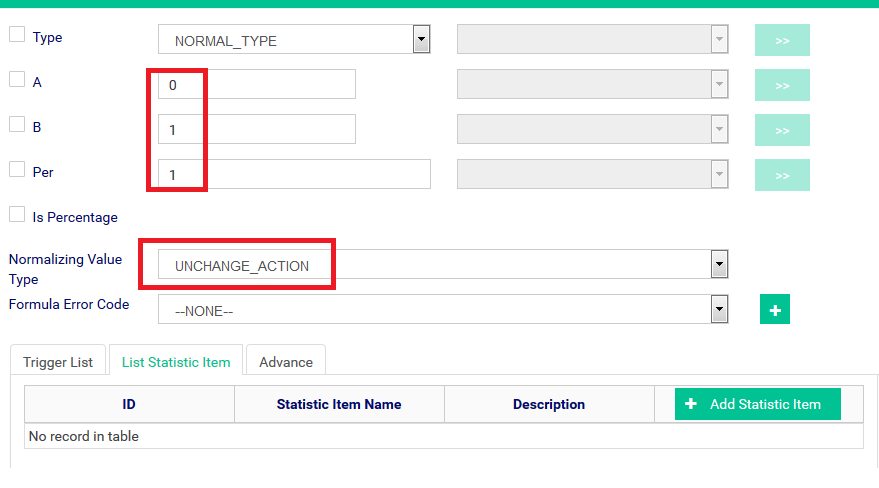
* Khai báo DT - Check VTfree reach Threshold 3 call sử dụng Rate trên



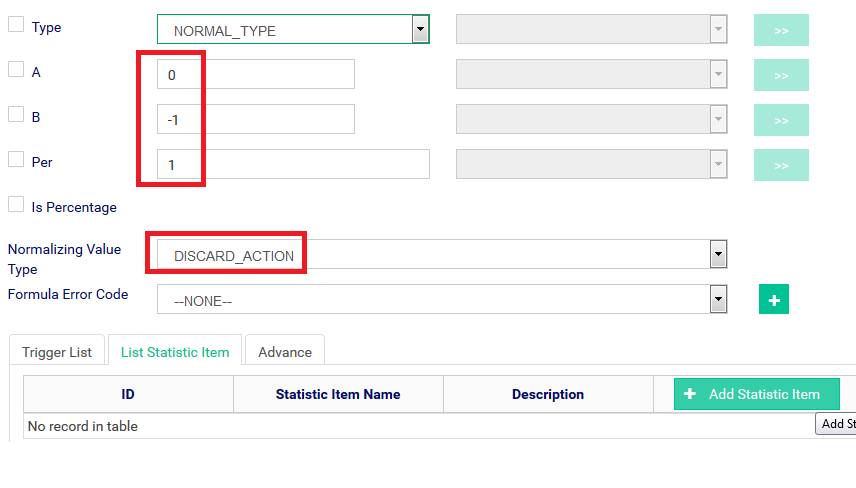
* Khai báo RT - Accumulate VTFREE reach to 3 call (offer not exist Skip, else add block)



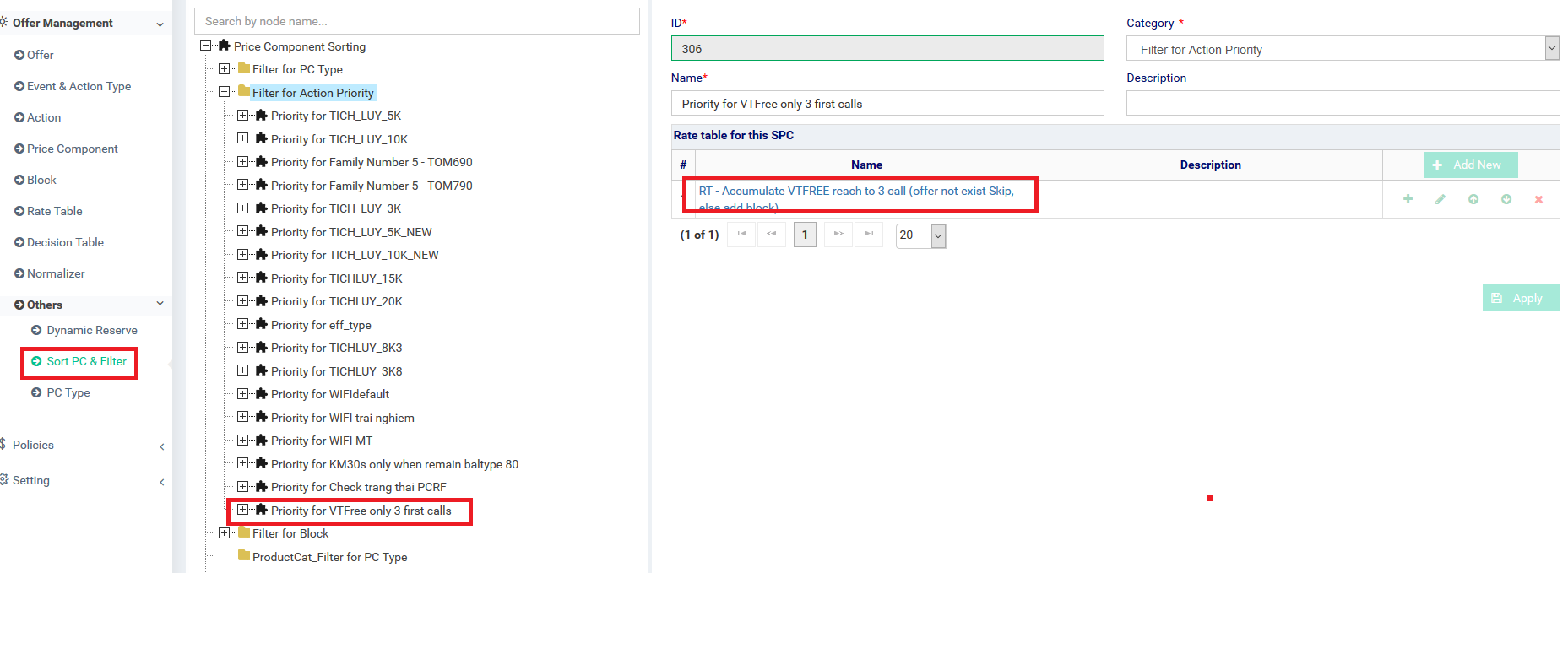
+ Formula của giá trị 1 là



+ Formula thứ 2 là



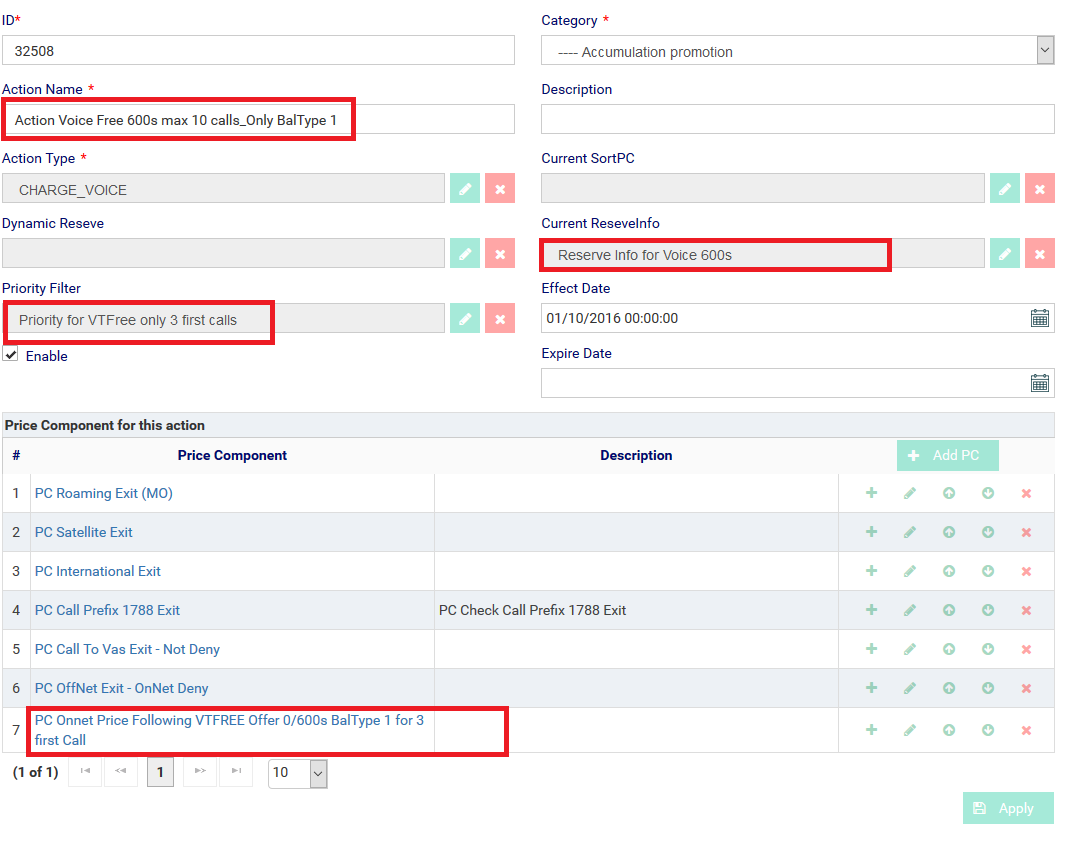
* Tạo ***Priority for VTFree only 3 first calls*** sử dụng Rate trên



## **Khai báo Action tính cước bằng cách clone từ Action Voice TICH\_LUY\_3K\_Only BalType 1**

Sau khi clone,

* Đổi tên Action: Action Voice Free 600s max 10 calls\_Only BalType 1
* Thực hiện Add Priority Filter ***“Priority for VTFree only 3 first calls”***
* Thực hiện thêm Current ReseveInfo
* Thực hiện xóa [PC Onnet Price For TICHLUY(Exist TIEUDUNG10P: 0/600s, else Exit) BalType 1](http://10.60.145.10:8080/vocs1/pages/tree_offer.xhtml?javax.faces.Token=eBWZVYG4YH6kbMWBwA%3D%3D&treeType=OFFER_ACTION)
* Thực hiện thêm PC Onnet Price Following VTFREE Offer 0/600s BalType 1 for 3 first Call



## **Khai báo thêm action reset tài khoản tích lũy 125 khi hủy gói cước**