Qu’est-ce qu’un autocom :

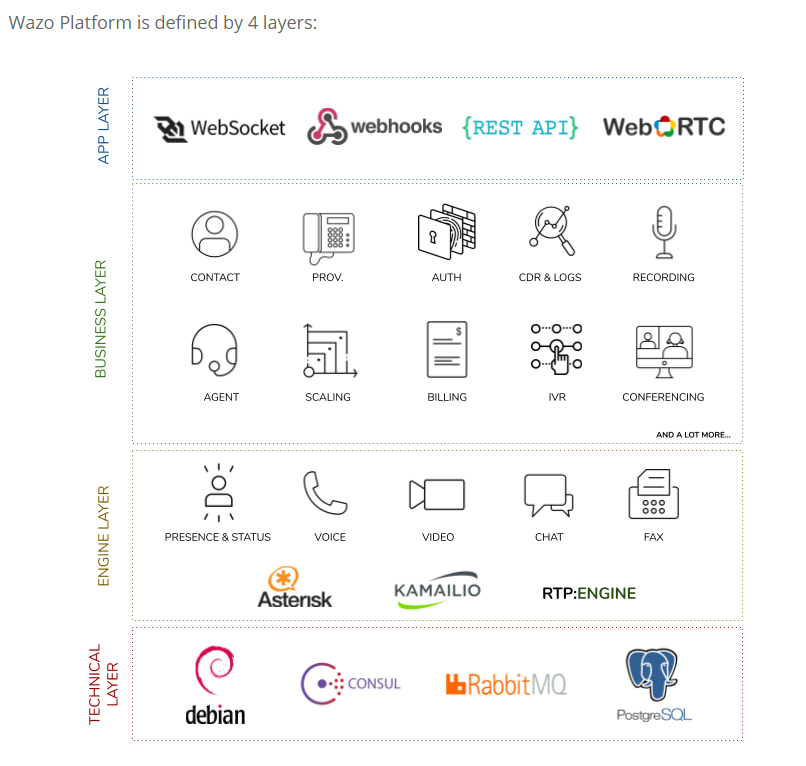
Un autocom est un appareil téléphonique sophistiqué qui permet de placer des appels en attente, de transférer des communications d’un poste à un autre, ou de diffuser des messages téléphoniques.

Pour la gestion des autocom Alliance Informatique utilise la plate-forme Wazo.

Wazo est un projet Open Source permettant de construire des infrastructures programmables de communication IP avec des features de classe 5 permettant les appels audio et vidéo, le chat, les centres d’appels, les conférences téléphoniques, les messages enregistrés….

Ils fournissent aussi des features de classe 4 comme la sécurité, le routing, la balance de chargement. On peut choisir les features qu’on désire et les intégrer à nos propres infrastructures existantes.

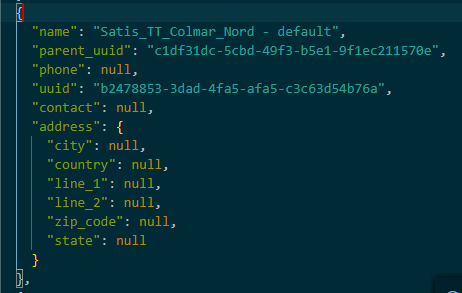
Nous utiliserons la « Business Layer »



Ici on utilise l’API RestFull de Wazo pour récupérer la liste des communications entrantes sortantes internes ou digitales, autrement appelé CDR (Call Détails Record) en format Json.

* Dans l’API Wazo une fois entrée notre clé de sécurité et notre header afin d’identifier notre entreprise
* Partie à étoffer la… pour montrer comment on s’identifie
* Nous récupérons en réalité 3 documents avec un get/List CDR

Le premier document que j’ai dénommé UserList (visible en annexe) contient le nom par défaut de la société qui possède la station d’appels, l’UUID qui est son identifiant unique au sein de Wazo, et le parent\_uuid qui est l’identifiant de la société qui a mis en place et s’occupe de la gestion des appels et leurs facturation en l’occurrence il s’agit ici d’Alliance Informatique.



On reçoit aussi la KeyList (visible en annexe) qui contient les SDA pour chaque UUID ainsi que les labels de clés qui permettront de savoir s’il s’agit d’un appel entrant sortant ou interne ou d’une communication par internet.

Un SDA est un acronyme pour sélection directe à l’arrivée. La SDA désigne une spécificité du réseau [RNIS](https://www.sewan.fr/lexique/rnis/) (lignes analogiques) qui permet d’appeler directement un poste interne (à une entreprise, une administration) avec un téléphone externe sans intermédiaire. On parle aussi de numéro direct. Par abus de langage, on appelle SDA tous les numéros de téléphone.

Le nom de chacune de ces clés est unique et en correspondance avec le ‘’tenant\_uuid’’ de la société utilisant l’autocom.



Il s’agit donc ici de mettre en relation en base de données chacune de ces clés correspondantes à l’uuid du client pour pouvoir ensuite les utiliser lors de l’extraction de données du CDR.

* Faire une grosse phase explicative sur les bdd et voir comment gérer le truc proprement pour ne pas réécrire 15000 fois la même chose
* Pensez à mettre des img ds le dossier pr illustrer tout ça

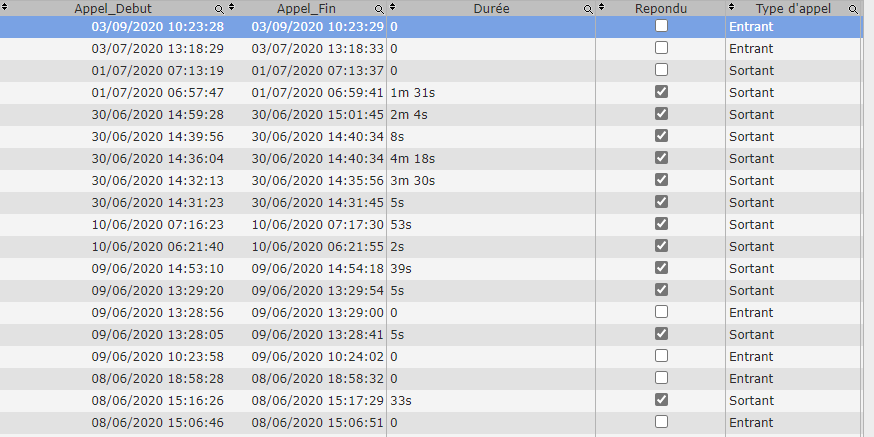
J’ai donc créé deux procédures ADD\_Clients\_UUID () et ADD\_Key\_NDI\_SDA () (disponible en annexe) permettant tout d’abord d’entrer les UUID des clients dans a base de données si ils n’y existent pas déjà puis d’ajouter chaque clé selon son label (interne, externe, entrant).

Il s’agit maintenant d’extraire les données du CDR Wazo en faisant des boucles sur le nœud item en ne récupérant pas les appels internes et en extrayant les noms de sociétés correspondant dans la base de données. Le but étant d’afficher dans un tableau la liste d’appel du client lorsque celui-ci se logue.

Lors du log on passe l’UUID du client en globale puis :



Ce qui nous donne ce résultat dans le tableau qui sera fourni au client en temps presque synchrone, WAZO faisant des mises à jour de son API toutes les 10 minutes :



Une fois cette première partie Login client et affichage de son activité faite.

J’ai récupéré les CSV des différents opérateurs de services VoIP (Voice Over IP) avec qui nous sommes en relations et qui effectuent la prise en charge et la mise en relation entre appelant et appelé.

La première difficulté était que chaque fournisseur à sa manière d’identifier le client, d’identifier l’appel et de créer son CSV (visible en annexe).

Il a fallu donc créer des procédures pour récupérer dans le même format les données dans le même format les données selon les fournisseurs (Diatem, 2Diatem, Eurydice, Telecom Object). En effet il faut pouvoir afficher dans un tableau géré par l’administrateur l’intégralité des appels gérés par les fournisseurs pour vérification des données et simplifiés les recherches des appels. Vous trouverez les procédures utilisées en masque de saisie en annexe.

Toujours dans la partie gestion administrateur des appels. J’ai créé une page ou l’on relie le CDR Wazo et les CDR fournisseurs. Après discussion il a été décidé de ne prendre en compte que les Fournisseur Diatem et Télécom Object. En effet à partir de novembre ce seront les seuls fournisseurs pertinents pour la gestion des appels sortant des clients d’Alliance Informatique.

J’ai donc du créé une clé unique avec le CDR de Wazo pour chaque appel puis généré la même clé pour chaque appel enregistré par les fournisseurs dans leurs fichier csv et établir des comparaisons.

Il s’agissait de prendre la date en format AAAAMMJJ concaténé avec l’heure et les minutes (les secondes n’étant pas disponible pour le fournisseur Diatem et sur certains des CSV de Télécom Object) et les 9 derniers chiffres du numéro de destinataires.

J’avais 232 appels sortant enregistrés par Wazo sur la période recherché.

La première version de clé unique au format AAAAMMJJ+HHMM+Destinataire a obtenu 220 correspondance sur 232 cependant certains appels n’avaient pas la bonne correspondance de durée fourni en secondes par le CDR Wazo.

J’ai donc créé une deuxième version au format AAAAMMJJ+HHMM+Durée+Destinataire. J’ai obtenu 200 correspondances sur 232. En effet le problème étant que Wazo comptait dans la durée le temps de mise en relation avec le destinataire alors que les fournisseurs nous fournissent la durée exacte du temps d’appel.

J’ai donc dans une troisième version créé une fourchette de durée en deuxième condition après la comparaison des clés. Cette fourchette demande après vérification de la clé à ce que la durée d’appel du fournisseur soit comprise entre -1secondes de la durée de l’appel Wazo et +2 secondes de la durée de l’appel Wazo. J’ai aussi supprimé les minutes dans la clé unique pour éviter les problèmes d’arrondi à la minute supérieur qui n’est pas géré de la même manière selon les fournisseurs, et chez Telecom Object cela peut changer selon les périodes sans raison explicite à ce sujet. J’étais donc sur un format :

AAAAMMJJ+HH+Destinataire,

Puis conditionnelle :

Durée Wazo +2 <= Durée en secondes fournisseur >=Durée Wazo -1

J’ai obtenu 217 correspondances sur 232 résultats.

J’ai décidé pour plus de précision de réduire la tranche de précision horaire à la dizaine de minutes près avec un tronquage des données. La quatrième version était donc au format :

AAAAMMJJ+HHM+Destinataire,

Puis conditionnelle :

Durée Wazo +2 <= Durée en secondes fournisseur >=Durée Wazo -1 j’ai obtenu 217 correspondances pour 232 résultats

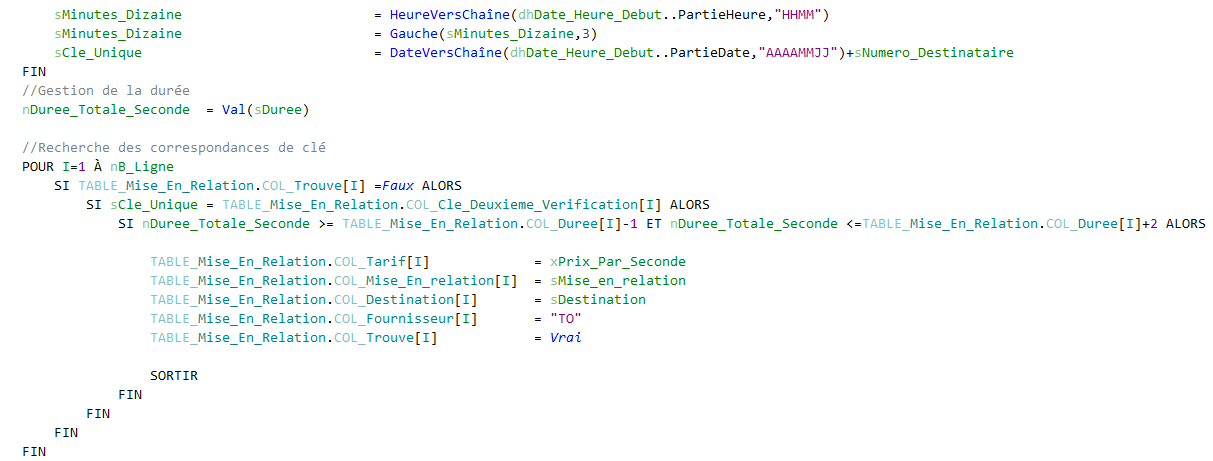
Après études des 15 résultats restants j’ai remarqué que Telecom Object ne donnait pas toujours le même format à ses csv parfois avec guillemets parfois sans, parfois le format heure au format HH:MM:SS parfois au format HH:MM. J’ai donc du affiner ma récupération de données de leur csv en ajoutant des conditionnels prenant en compte la taille ou les différentes écritures de dates. La cinquième version au format :

Vérification et tronquage selon la taille de la chaine récupéré  
génération d’une clé unique au format : AAAAMMJJ+HHM+Destinataire,

Puis conditionnelle : Durée Wazo +2 <= Durée en secondes fournisseur >=Durée Wazo -1



J’ai ainsi obtenu 226 correspondances pour 232 résultats.  
Sur les 6 appels non trouvés 5 n’existaient tout simplement pas dans mes fichiers csv fournisseurs. En effet je travaille avec des fichiers d’exemples et pas des données en temps réels le temps de trouver une c lé optimale sans abimer de la donnée non traitée à l’heure actuelle et afin de ne pas gêner les facturations en cours sur le mois de septembre. Le dernier résultat est un appel qui a été passé selon le CDR Wazo à 14h39m56sec. Il a été pris en compte par le fournisseur Telecom Object à 14h40. Faire un format horaire arrondi pouvant amener à plus d’erreur j’ai décidé de faire une deuxième vérification sur les correspondances manquantes qui aurait une clé secondaire au format : AAAAMMJJ+ Destinataire   
puis conditionnelle de Durée.



Cette version simplifiée m’a permis de retrouver les appels manquants du côté fournisseur.

Pour être sure de gérer tous les cas j’ai ensuite instauré une procédure qui récupère les appels du CDR Wazo qui n’ont pas trouvé de correspondance fournisseur et qui crée un fichier au format CSV dans un chemin de fichier pour l’instant provisoire mais qui sera défini lors de l’implémentation finale pour vérifier manuellement en cas d’appel non trouvé. Ce qui nous donne au final :

nResultats est un entier

nB\_Lignes est un entier

I est un entier

sChemin\_Fichier est une chaîne

nInd\_Fichier est un entier

sLigne\_Rapport est une chaîne

sContenu\_Rapport est une chaîne

sChemin\_Fichier = "C:\Users\Audrey\Desktop\Téléphonie\Appel\_Non\_Trouve-"+DateHeureSys()+".csv"

sDate\_Heure\_Appel est une chaîne

dhDateHeure est une DateHeure

Diatem\_Correspondance()

T\_O\_Correspondance()

nResultats = TableCherche(COL\_Trouve,*Faux*)

nB\_Lignes = TableOccurrence(TABLE\_Mise\_En\_Relation)

SI nResultats <>-1 ALORS

Deuxieme\_Verification\_Diatem()

Deuxieme\_Verification\_T\_O\_Correspondance()

nResultats = TableCherche(COL\_Trouve,*Faux*)

SI nResultats<>-1 ALORS

sLigne\_Rapport= "Heure\_d'appel\_Debut;Numero\_Appele;Duree;"

sContenu\_Rapport+= sLigne\_Rapport+*RC*

POUR I = 1 À nB\_Lignes

SI TABLE\_Mise\_En\_Relation.COL\_Trouve[I]=*Faux* ALORS

dhDateHeure = COL\_Date\_Debut[I]

sDate\_Heure\_Appel= DateVersChaîne(dhDateHeure..PartieDate,"JJ/MM/AAAA")+" "+ HeureVersChaîne(dhDateHeure..PartieHeure,"HH:MM:SS")

sLigne\_Rapport=sDate\_Heure\_Appel+";"+Droite(COL\_Cle\_T\_O[I],9)+";"+COL\_Duree[I]

sContenu\_Rapport+= sLigne\_Rapport+*RC*

FIN

FIN

fSauveTexte(sChemin\_Fichier,sContenu\_Rapport)

FIN

FIN

Vous trouverez les procédures Diatem\_Correspondance ( ), T\_O\_Correspondance ( ), Deuxieme\_verification\_Diatem ( ) et Deuxieme\_Verification\_T\_O\_Correspondance ( ) en annexe