# Raciocínio de Status

df\_conf é a listagem mestra, tem “todos os status”

compara df\_conf com o df\_ativ (todas as placas que estão em um conjunto com status ATIVO e o benefício com o status ATIVO)

comparação com todas (pega todos os registros) e vê o penúltimo:

- primeiro vê se o penúltimo status do benefício não era CANCELADO (ou semelhante), pois agora está ATIVO então será REATIVAÇÃO, caso contrário:

- se tiver empresa diferente = MIGRAÇÃO (ST - VIA)

- se for a mesma = RENOVAÇÃO (ST – ST)

- e o restante, como df\_ativ é de ontem é NOVO, esse status aqui então se refere às placas que foram ativadas hoje sem histórico prévio

MUDANÇAS E PENSAMENTOS E PREMISSAS

- mesmo que veículos estejam na granularidade de coverage benefícios, se uma placa é cancelada ou ativada, todos os seus benefícios também o são

- para fins de histórico tenho que considerar o conjunto, pois se um benefício de uma placa for cancelado sendo que o outro benefício dessa mesma placa está ativo ela não está cancelada! (assim é feito erroneamente no board\_status\_treatment para considerar reativações de placas) DEVO FAZER PELO CONJUNTO

- conclusão importante: mesmo que esteja na granularidade do benefício, devo eliminar duplicatas de placa avaliando se tem alguma com status\_benefício ATIVO

- isso aqui me intrigou para considerar RENOVAÇÃO

irs.id\_renovated\_set as "id\_renovacao", mas é que ele mostra que a origem dele veio de uma renovação, algo no passado, isso não me ajuda mto

- atentar para o fato de que quando muda o status o conjunto tende a mudar

- o status benefício pode ficar ATIVO com o conjunto já FINALIZADO

- a proposta nova é fazer incremental, compondo sempre pelo último registro de ativos

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Aparentemente quando há cancelamento há mudança de conjunto

Data\_registro é a data em que o conjunto (que muda constantemente) foi registrado (pouco antes da ativação)

Pelo que to vendo eles não cancelam a placa pra iniciar em outra empresa

Retirar duplicatas de ativações e cancelamentos de ontem **(aqui pode haver perda de infos reavaliar depois)**

depois acrescentar migration\_from

# Resumo do código

- (1) data\_frame do dia anterior (df\_aggregate\_yesterday)

- (2) data\_frame base (histórico até do que já foi acrescentado previamente) – é o template

- (1) & (2) ambos vc separa cancelados e ativos (no esquema abaixo), concatena-os e cria uma nova coluna de vigência e data\_status

- OBS: vigência é ATIVO (NOVO, ATIVO, RENOVAÇÃO) ou CANCELADO (CANCELADO, FINALIZADO, SIMILARES)

- cria uma coluna de status e migration\_from e adiciona no yesterday

- faz o tratamento do status

- todas as placas movimentadas terão um histórico associado a partir do dia um

- DEIXAR O ALL ESTÁTICO

RESULTADOS:

Na coluna de vigência é possível pegar todos os ativos, que não estão cancelados

Na de status pegar último status de placa

Obs: tem alguns que tem ativo e data de cancelamento depois mas estão como renovação antes por isso retirei os data\_cancelamento = null