

Programul de fidelizare Partener Colibri

Data mapping

A Colibri Partner is:

```
mantle.party.PartyRole
partyId = partner.id
roleTypeId = "Affiliate"

mantle.party.PartyIdentification
partyId = partner.id
partyIdTypeEnumId = "PtidAffiliateId"
idValue = partnerCode

mantle.party.contact.ContactMech
contactMechId = partner.id + "_PHONE"
contactMechTypeEnumId = "CmtTelecomNumber"
mantle.party.contact.TelecomNumber
contactMechId = contactMechId
contactNumber = phone
mantle.party.contact.PartyContactMech
partyId = partner.id
contactMechId = contactMechId
contactMechPurposeId = "PhonePrimary"
fromDate = now

mantle.account.financial.FinancialAccount
finAccountId = partner.id
finAccountTypeId = "ServiceCredit"
statusId = "FaActive"
finAccountName = "Partener Colibri"
organizationPartyId = L2
```

```
ownerPartyId = partner.id
```

```
mantle.party.agreement.AgreementParty
```

```
agreementId = "PartenerColibri"
```

```
partyId = partner.id
```

```
roleTypeId = "Affiliate"
```

```
mantle.party.agreement.Agreement
```

```
agreementId = "PartenerColibri"
```

```
agreementTypeEnumId = "AgrCommission"
```

```
organizationPartyId = L2
```

```
organizationRoleTypeId = "OrgInternal"
```

```
mantle.party.agreement.AgreementTerm
```

```
agreementTermId = "PartenerColibri_1..10"
```

```
agreementId = "PartenerColibri"
```

```
termTypeEnumId = "TtCommission"
```

```
termNumber = 50,150,300,500,750,1500,2500,3500,6000,10000
```

```
minQuantity = 500,5000,10000,15000,20000,30000,40000,50000,75000,100000
```

A received payment from a Colibri partner is:

```
mantle.account.payment.Payment
```

```
paymentId = legacyIncasare.id
```

```
paymentTypeEnumId = "PtInvoicePayment"
```

```
fromPartyId = legacyIncasare.partner.id
```

```
toPartyId = L2
```

```
statusId = "PmntDelivered"
```

```
effectiveDate = legacyIncasare.dataDoc
```

```
amount = legacyIncasare.getTotal()
```

```
mantle.account.payment.PaymentParty
```

```
paymentId = paymentId
```

```
partyId = affiliatePartnerId
```

```
roleTypeId = "Affiliate"
```

When a new threshold is reached the customer credit is replenished like this:

```
mantle.account.financial.FinancialAccountTrans
```

```
finAccountTransId = "PC${partner.id}_1..10"
```

```
finAccountTransTypeEnumId = "FattDeposit"
finAccountId = partner.id
fromPartyId = L2
toPartyId = partnerAffiliate.id
transactionDate = now
amount = termNumber - previous termNumber
```

Service mapping

WHEN

```
foreach incaseazaDocsOrPartner().paidDocs AS legacyIncasare
closeBonCasa().get(InvocationResult.ACCT_DOC_KEY) AS legacyIncasare
closeFacturaBCAviz().get(InvocationResult.CHITANTA_KEY) AS legacyIncasare
```

THEN

```
createPayments()
```

```
seca: checkPartnerAgreements after createPayments()
```

```
affiliatePartyId = PaymentParty.partyId WHERE PaymentParty.roleTypeId = "Affiliate"
```

```
if null affiliatePartyId = Payment.fromPartyId
```

```
agreementParty = AgreementParty WHERE agreementId = "PartenerColibri" AND roleTypeId =
"Affiliate" AND partyId = affiliatePartyId
```

```
if agreementParty is null then EXIT
```

```
lastThresholdReached = FinancialAccountTrans WHERE toPartyId = affiliatePartyId AND
finAccountTransTypeEnumId = "FattDeposit" AND fromPartyId = "L2" AND finAccountId =
affiliatePartyId ORDER BY finAccountTransId.split("_")[1].toInt GET LARGEST
```

```
nextThreshold = AgreementTerm WHERE agreementTermId = "PartenerColibri_${lastThresholdIndex +
1}" ?: "PartenerColibri_1"
```

```
lastReachedTerm = AgreementTerm WHERE agreementTermId =
"PartenerColibri_${lastThresholdIndex}" ?: null
```

```
if nextThreshold is null then EXIT
```

```
if affiliatePartyPaymentsTotal > nextThreshold.minQuantity AND NOT EXISTS
```

```
FinancialAccountTrans WHERE finAccountTransId = "PC${partner.id}_${lastThresholdIndex + 1}"
```

```
create FinancialAccountTrans
```

```
    finAccountTransId = "PC${affiliatePartyId}_${lastThresholdIndex + 1}"
```

```
    finAccountTransTypeEnumId = "FattDeposit"
```

```
    finAccountId = partner.id
```

```
    fromPartyId = "L2"
```

```
    toPartyId = affiliatePartyId
```

```
transactionDate = now
amount = nextThreshold.termNumber - (lastReachedTerm.termNumber ?: 0)
create legacy DiscountDoc incasare
send threshold reached SMS to affiliate
else
send payment received SMS to affiliate
```

Revision #15

Created 2026-04-22 09:18:57 UTC by Groza Danut

Updated 2026-04-28 11:30:32 UTC by Groza Danut