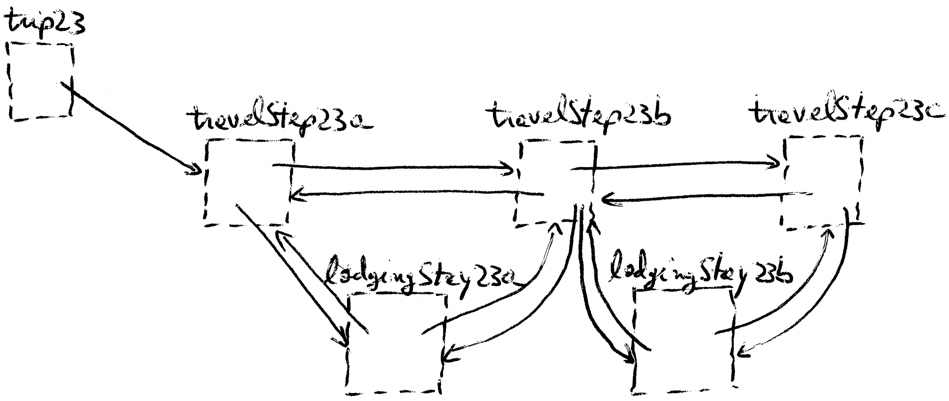
Reprezentarea orientată obiect - Un exemplu

O bază de cunoștințe cu cadre poate fi văzută ca o foaie de calcul simbolic, constrângerile între obiecte fiind propagate în KB de procedurile atașate.

Exemplu de planificare a călătoriilor

Începem cu o schiță a structurii unei călătorii



Avem 2 cadre principale Trip și TravelStep

Un Trip va avea o secvență de TravelStep. Un TravelStep se termină de obicei într-un LodgingStay, cu excepția a 2 etape parcurse în aceeași zi sau a etapei finale din călătorie.

LodgingStay $\xrightarrow{\text{slot-uri}}$ arrivingTravelStep
→ departingTravelStep

TravelStep → originLodgingStay
→ destinationLodgingStay

Cadrul Trip poate arăta astfel:

```
(Trip
    <:FirstStep   TravelStep>
    <:Traveler    Person>
    <:BeginDate   Date>
    <:EndDate     Date>
    <:TotalCost   Price>
   ...)
```

```
(trip23
  <:INSTANCE-OF Trip>
  <:FirstStep travelStep23a
  <:Traveler  davidF>
  <:BeginDate 23/11/18>
  <:EndDate  25/11/18>
  <:TotalCost 3400RON>
 ...)
```

Cadrele TravelStep și LodgingStay împart o serie de proprietăți pe care le grupăm în cadrul mai general TripPart

```
(TripPart
    <: BeginDate  Date >
    <: EndDate  Date >
    <: Cost  Price >
    <: PaymentMethod  FormOfPayement >
   ...)
(LodgingStay
    <: IS-A  TripPart >
    <: Place  City >
    <: LodgingPlace  LodgingPlace >
    <: ArrivingTravelStep  TravelStep >
    <: DepartingTravelStep  TravelStep >
   ...)
(TravelStep
    <: IS-A  TripPart >
    <: Origin  City >
    <: Destination  City >
    <: OriginLodgingStay  LodgingStay >
    <: DestinationLodgingStay  LodgingStay >
    <: Means  FormOfTransportation >
    <: DepartureTime  Time >
    <: ArrivalTime  Time >
    <: NextStep  TravelStep >
    <: PreviousStep  TravelStep >
   ...)
```

Filler-e implicite

```
(TravelStep
      <: Means airplane > ...)
(TripPart
    <: PaymentMethod  VisaCard > ...)
(TravelStep
    <: PaymentMethod  masterCard > ...)
```

Notatii: - daca x este un cadru individual si y este o referinta catre un slot, xy este filler-ul acelui slot din cadru.

-SELF - referinta catre cadrul curent procesat.

Filler-ul slot-ului :Origin din TravelStep poate fi stabilit/calculat astfel:

```
(TravelStep
    <:Origin  [IF-NEEDED
                  {if no SELF:PreviousStep then otepeni;
                      else SELF:PreviousStep:Destination}
              ] > ...)

(Trip
    <:TotalCost [IF-NEEDED
                    {result ← 0
                     x ← SELF:FirstStep;
                     repeat
                       if exists x:NextStep then
                           {result ← result + x:Cost
                            if exists x:DestinationLodgingStay then
                               {result ← result + x:DestinationLodgingStay:Cost;
                            x ← x:NextStep;
                           }
                       else return result + x:Cost;
                    }
                ] > ...)

(TravelStep
    <:NextStep [IF-ADDED
                   {if SELF:EndDate ≠ SELF:NextStep:BeginDate then
                       {SELF:DestinationLodgingStay ←
                        SELF:NextStep:OriginLodgingStay ←
                           new LodgingStay
                              with :BeginDate = SELF:EndDate;
                              with :EndDate = SELF:NextStep:BeginDate
                              with :ArrivingTravelStep = SELF;
                              with :DepartingTravelStep = SELF:NextStep;
                       }
                   }
               ] > ...)
```

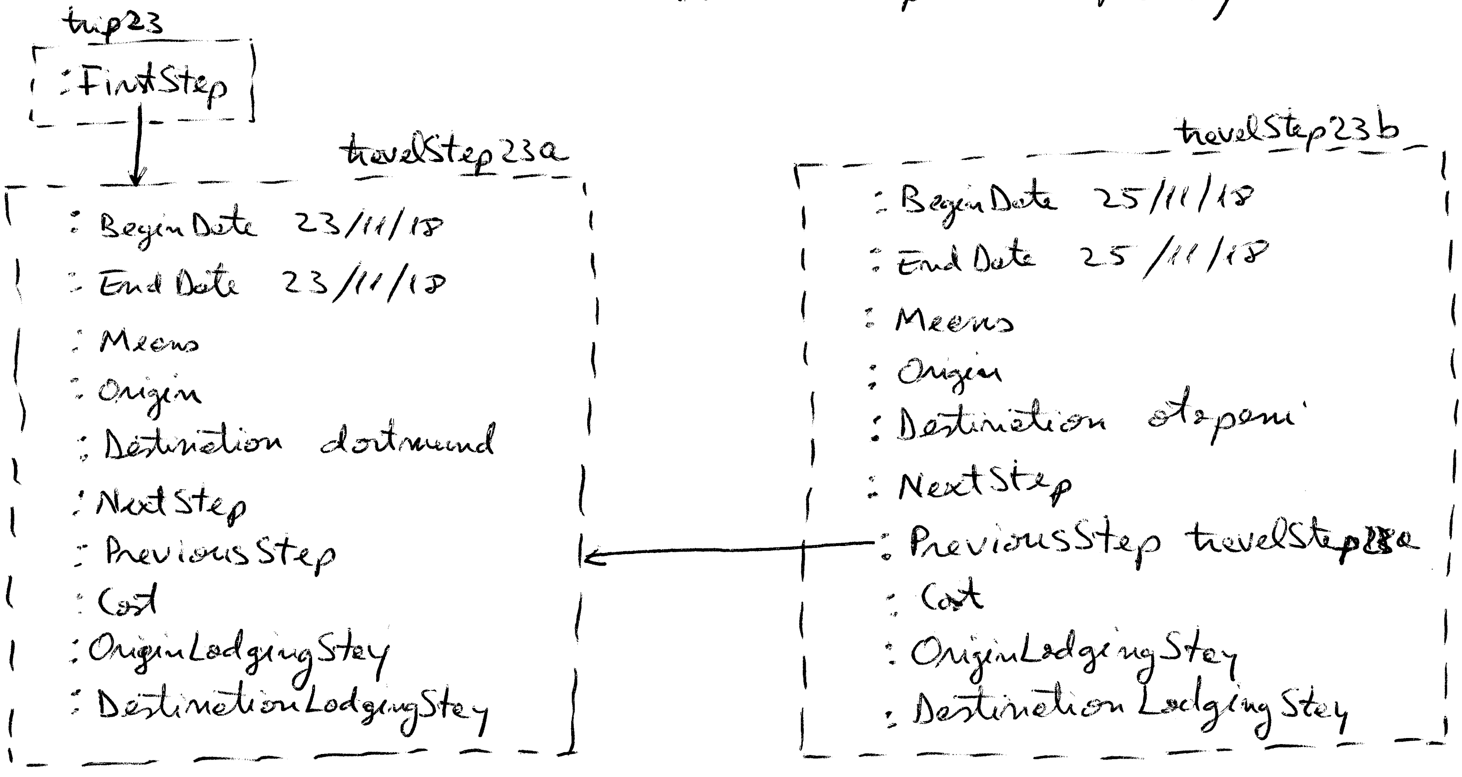
```
(LodgingStay
    <:Place [IF-NEEDED
              {SELF : ArrivingTravelStep : Destination}
            ] > ...)
```

Pentru o călătorie anume (trip23) creăm un cadru individual

```
(trip23
    <:INSTANCE-OF Trip>
    <:FirstStep travelStep23a >)
```

şi un cadru (travelStep23a
<:INSTANCE-OF TravelStep >
<:Destination dortmund >
<:BeginDate 23/11/18 >
<:EndDate 23/11/18 >)

Creăm un al doilea cadru (travelStep23b
<:INSTANCE-OF TravelStep>
<:Destination otopeni >
<:BeginDate 25/11/18 >
<:EndDate 25/11/18 >
<:PreviousStep travelStep23a >)



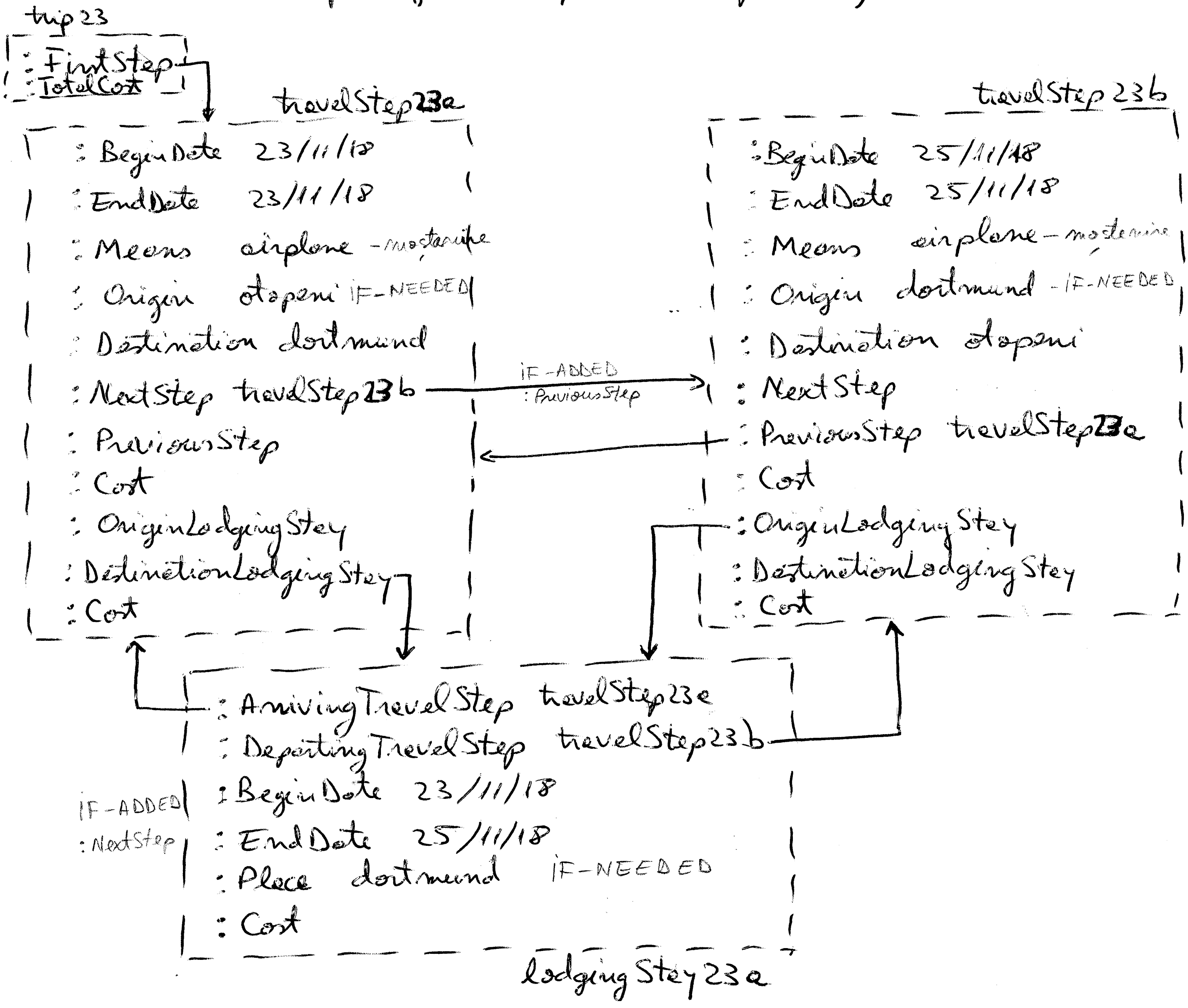
Pentru a încheia inițializarea trebuie ca travelStep18a :NextStep ← travelStep18b

```
(TravelStep
        <:PreviousStep [IF-ADDED
                {SELF:PreviousStep:NextStep ← SELF;}
              ]>...)
```

Prin această atribuire se execută procedura IF-ADDED atașată slot-ului :NextStep

```
(lodgingStay23a
        <:INSTANCE-OF LodgingStay>
        <:BeginDate 23/11/18>
        <:EndDate 25/11/18>
        <:ArrivingTravelStep travelStep23a>
        <:DepartingTravelStep travelStep23b>)
```

dacă este cerut, slot-ul :Place va avea filler-ul dortmund prin procedura atașată IF-NEEDED.


trip23
:FirstStep
:TotalCost
travelStep23a
:BeginDate 23/11/18
:EndDate 23/11/18
:Means airplane -moștenire
:Origin otopeni IF-NEEDED
:Destination dortmund
:NextStep travelStep23b
:PreviousStep
:Cost
:OriginLodgingStay
:DestinationLodgingStay
:Cost
IF-ADDED
:PreviousStep
travelStep23b
:BeginDate 25/11/18
:EndDate 25/11/18
:Means airplane -moștenire
:Origin dortmund -IF-NEEDED
:Destination otopeni
:NextStep
:PreviousStep travelStep23a
:Cost
:OriginLodgingStay
:DestinationLodgingStay
:Cost
:ArrivingTravelStep travelStep23a
:DepartingTravelStep travelStep23b
:BeginDate 23/11/18
:EndDate 25/11/18
:Place dortmund IF-NEEDED
:Cost
IF-ADDED
:NextStep
lodgingStay23a