Fakultet tehničkih nauka
Inženjerstvo informacionih sistema

Predmet: Projektovanje skladišta podataka

Projekat

Data Warehouse za analizu ekspedicija na Himalaje

Student: Kosta Bjelogrlić IT31-2021

Sadržaj

1. Zadatak i ciljevi projekta

Glavni zadatak ovog projekta jeste razvijanje centralizovanog izvora informacija o ekspedicijama na Himalaje. Taj centralizovani izvor informacija će poslije služiti kao osnova za pouzdane analitičke izvještaje. Osnova za izradu ovog projektnog zadatka je *Himalayan Expeditions* javno dostupni *dataset*, koji u sebi sadrži sve zabilježene podatke o poznatim ekspedicijama na Himalaje u periodu od 1900 do 2024 godine. Upravo zbog masivnosti i nivoa detaljnosti ovoga *dataset*-a, on predstavlja dobru osnovu za kvalitetnu transformaciju podataka u skladište podataka koje služi za bolje i preciznije donošenje zaključaka o samim ekspedicijama. Cilj ovog projektnog zadatka jeste da se sirovi i fragmentisani podaci ovog *dataset*-a, koju su zabilježeni u *CSV* fajlovima, prebace odnosno transformišu u skladište podataka. Unutar njega svi podaci će biti organizovani, očišćeni i standardizovani na takav način da će omogućiti brzu, jasnu i efikasnu analizu. Ovim pružamo mogućnost za lako i brzo kreiranje velikog broja pouzdanih izvještaja. Korisnicima će biti omogućeno da kroz uvid u izvještaje dobiju precizan i jasan uvid za sve informacije koje su od njihovog interesa na jednom mjestu. Polazni javno dostupan *dataset* je dat u formi *OLTP* šeme i zadatak jeste bio da se na osnovu potreba korisnika identifikuje šta je to od podataka što interesuje korisnika, i da se na osnovu identifikovanih potreba i podataka deviniše jasna *OLAP* šema koja će kasnije predstavljati skladište podataka koje će u sebi sadržati podatke neophodne za izradu izvještaja. Sama *OLAP* šema će biti u formi zvjezdaste šeme, koja će sadržati jasno definisane dimenzione tabele i jednu činjeničnu tabelu. Granularnost činjenične tabele će predstavljati učestvovanje jednog člana u određenoj ekspediciji. U toj činjeničnoj tabeli na osnovu zadatog nivoa granularnosti svaki red će predstavljati učešće jednog člana u određenoj ekspediciji sa svim njegovim karakteristikama i mjerama (najveća dostignuta tačka u toj ekspedicji, da li je dostigao ciljni vrh ekspedicije, da li je koristio dodatni kiseonik tokom ekspedicije...). Zbog velikog nivoa detaljnosti polaznog *dataset*-a bilo je neophodno identifikovati koji će podaci biti neophodni za izradu izvještaja, a da se pri tome zadrži jednostavanost implementacije i analize podataka. Glavni cilj projektnog zadatka jeste da se na osnovu prethodno kreiranog skladišta podataka kreiraju izvještaji u *SSRS* koji će pružati odgovore na unaprijed definisana korisnička pitanja. Ovim projektnim zadatkom obuhvaćen je cjelokupan proces od kreiranja skladišta podataka, zatim ekstrakcije, transformacie i učitavanje podataka iz sirovih polaznih *CSV* fajlova u kreirano skladište podataka, do analize podataka iz skladišta i kreiranja izvještaja na osnovu sprovedenih analiza.

2. Opis postupka projektovanja DW sistema

Glavni zadatak ovog projektnog zadatka jeste bio kreiranje centralizovanog izvora svih informacija neophodnih za dalju analizu. Skladište podataka odnosno *Data Warehouse* predstavlja taj centralizovani izvor odnosno centralizovanu bazu podataka koja služi za skladištenje podataka i za sprovođenje analze nad tim podacima. Ovi *DW* sistemi se koriste za potrebe analize podataka i da se na osnovu sprovedenih analiza donose razne poslovne odluke. Dok sa druge strane standardne transakcione baze podataka su uglavnom fokusirane na operativno poslovanje.

Prvi korak u postupku projektovanje *DW* sistema bio je detaljan uvid i pregledanje svih fajlova u polaznom *dataset*-u. Ovaj korak je obuhvatao je razumjevanje strukture fajlova, načina na koju su fajlovi međusobno povezani, kakva obilježja postoje u tim fajlovima, kakva je njihova osnovna struktura i šta oni zaista predstavljaju. Zbog masivnosti i velikog nivoa detaljnosti ovog *dataset*-a ovaj korak je bio među najbitnijim i najtežim dijelovima u ovom projektnom zadatku, jer je duboko razumjevanje svih detalja na samom početku od velikog značaja za kasnije sprovođenje što preciznije analzie podataka. Dubokom analizom svih polaznih podataka utvrđeno je da je riječ o bogatom setu podataka i da bi se za granularnost trebalo uzeti pojedinačno učestvovanje jednog učesnika u određenoj ekspediciji. Zatim su u svim ovim podacima identifikovani samo oni podaci koji su neophodni za zadati nivo granularnosti. Identifikovani su pojedinačni učesnici na ekspedicijama, detalji o samim ekspedicijama, ciljni vrhovi svih ekspedicija, nacionalnost svih učesnika, njihov status u ekspedicji, rute po kojima su se kretali kao i razlozi zbog kojih su učesnici morali da napuste ekspediciju u kojoj su učestvovali. Na ovaj način svi ključni i neophodni podaci ostaju vezani za jedan događaj a da se pri tome zadrži preglednost i jednostavnost skladišta podataka. Dalje je odlučeno da se koristi zvjezdasta šema jer je ona pogodna i dobro prilagođenja za ovaj tip analitičkih sistema. Centralnu figuru ove zvjezdaste šeme predstavlja činjenična tabela čiji redovi predstavljaju učestvovanje jednog učesnika u određenoj ekspediciji i sadrži mjere (da li je taj učesnik bio lider u toj ekspediciji, da li je uspio sa svojom ekspedicijom da dođe do ciljnog vrha, da li je koristio dodatni kiseonik tokom ekspedicije, maksimalnu visinu koju je dostigao tokom te ekspedicije, kao i broj godina koje je učesnik imao tokom ekspedicije). Oko te činjenične tabele su postavljene dimenzione tabele koje su sve povezane sa centralnom činjeničnom tabelom putem stranih ključeva koji se nalaze u činjeničnoj tabeli. Od dimenzija imamo vremensku dimenziju, ekspediciju, učesnike, vrhove, statuse učesnika u ekspediciji, rute, nacionalnosti kao i razloge zašto su učesnici ranije završavali svoje ekspedicije. Sledeća faza nakon kreiranja pomenutog modela bila je implementacija *ETL* procesa u *SSIS*-u. Tu su svi neophodni podaci učitavani iz *CSV* fajlova i kasnije su prošli kroz čitav niz transformacija koje su obuhvatale čišćenje podataka, zamjena praznih vrijednosti ili *null* vrijednosti sa "*Unknown*" vrijednostima, zamjena nepoznatih brojčanih vrijednosti sa –1, zatim i formatiranje kolona u odgovarajuće formate. Nakon toga su svi neophodni podaci upisani u skladište podataka gdje su nad njima u finalnom koraku u *SSRS* sprovedene analize i napravljeni izvještaji na osnovu korisničkih zahtjeva

3. Specifikacija zahtjeva korisnika

Zahtjevi korisnika su predstavljeni u formi pitanja. Da bi jedan ovakav analitički sistem uopšte imao svrhu svog postojanja moramo imati neke korisničke zahtjeve odnosno pitanja na koja moramo dati odgovor i pružiti našim korisnicima uvid u sprovedene analitičke izvještaje.
Prvo pitanje se odnosi na uspješnost članova ekspedicije po njihovoj nacionalnosti. Korisnik želi da zna koje su to nacije najuspješnije na ekspedicijama, odnosno kojoj nacionalnosti pripadaju članovi ekspedicija sa najvećim procentom uspješnosti na ekspediciji, odnosno onda kada se popenju na ciljni vrh ekspedicije. Ovim izvještajem korisnik dobija jasnu informaciju koje su to nacije sa najvećim procentom uspješnih ekspedicija, odnosno one nacije koji se najbolje snalaze na Himalajima i koji su dobri planinari.
Drugo pitanje se odnosi na najčešće razloge zašto su članovi ekspedicija odustajali od daljeg putovanja tokom ekspedicije. Ovim izvještajem korisnik dobija uvid u to koji su najčešći razlozi zašto su pojedinačni članovi odustali od ekspedicije i nisu dalje nastavljali sa ostalim članovima već se vratili nazad u početni kamp. Na taj način korisnik može da identifikuje šta je to na šta treba da obrati pažnju tokom pripreme za ekspediciju i tokom same ekspedicije i na osnovu toga dobro da se pripremi i da uspije u svojoj ekspediciji. Takođe može i da vidi koje su česte vremenske neprilike koje ga mogu zadesiti i da na osnovu toga procjeni adekvatno godišnje doba za svoj poduhvat.
Treće pitanje se odnosi na najopasnije vrhove na Himalajima. Korisnik želi da zna koji su to vrhovi na Himalajima najopasniji i najrizičniji za osvajanje. Kao metriku za opasnost nekog vrha uzet je broj članova ekspedicije koji se nažalost nikada više nisu vratili sa pohoda na taj vrh. Korisnik će ovim izvještajem dobiti jasane podatke koji su to vrhovi najopasniji za osvajanje i u skaldu sa tim da se i adekvatno pripremi i organizuje za pohode na takve planinske vrhove.
Četvrto pitanje se odnosi na uspješnost članova ekspedicija po starosnim grupama. Korisnik želi da zna koji je to procenat uspješnosti članova ekspedicije, odnosno onih članova koji su se uspješno popeli na ciljni vrh te ekspedicije, po starosnim grupama od najmlađih pa sve do najstarijih. Ovim izvještajem korisnik ima uvid u to koje su se starosne grupe najbolje pokazale u pohodima na planinske vrhove.
Peto korisničko pitanje se odnosi na upoređivanje uspješnosti *Šerpasa* na ekspedicijama i ljudi ostalih nacionalnosti. Ovim izvještajem korsinik dobija jasan uvid u to kakav je odnos uspješnosti *Šerpasa* na ekspedicijama i ostalih ljudi drugih nacionalnosti, i na osnovu toga može da donese odluku da li da angažuje nekog *Šerpasa* da mu pomogne tokom pohoda ili da im da neke dodatne savjete ili preporuči neke rute kojima da se njegova ekspedicija kreće ka zacrtanom vrhu.
Šesto pitanje se odnosi na uticaj korišćenja dodatnog kiseonika na ishod člana ekspedicije, odnosno da li će uspjeti da se popenje na ciljni vrh ekspedicije ili ne ukoliko koristi dodatni kiseonik. Korisnik ovim izvještajem dobija jasan uvid u to kako utiče korišćenje dodatnog kiseonika na ispod ekspedicije i na taj način može da donese odluku o snadbjevanju sa dodatnim kiseonikom i korišćenjem istog tokom svoje ekspedicije.
Sedmo pitanje se odnosi na broj ekspedicija po decenijama. Korisnik želi da zna kada su tokom istorije ekspedicije na Himalaje bile popularne i koliko je bilo ekspedicija tokom istorije u određenoj dekadi. Ovim izvještajem korisnik dobija jasan uvid u broj ekspedicija po dekadama.

4. Specifikacija modela

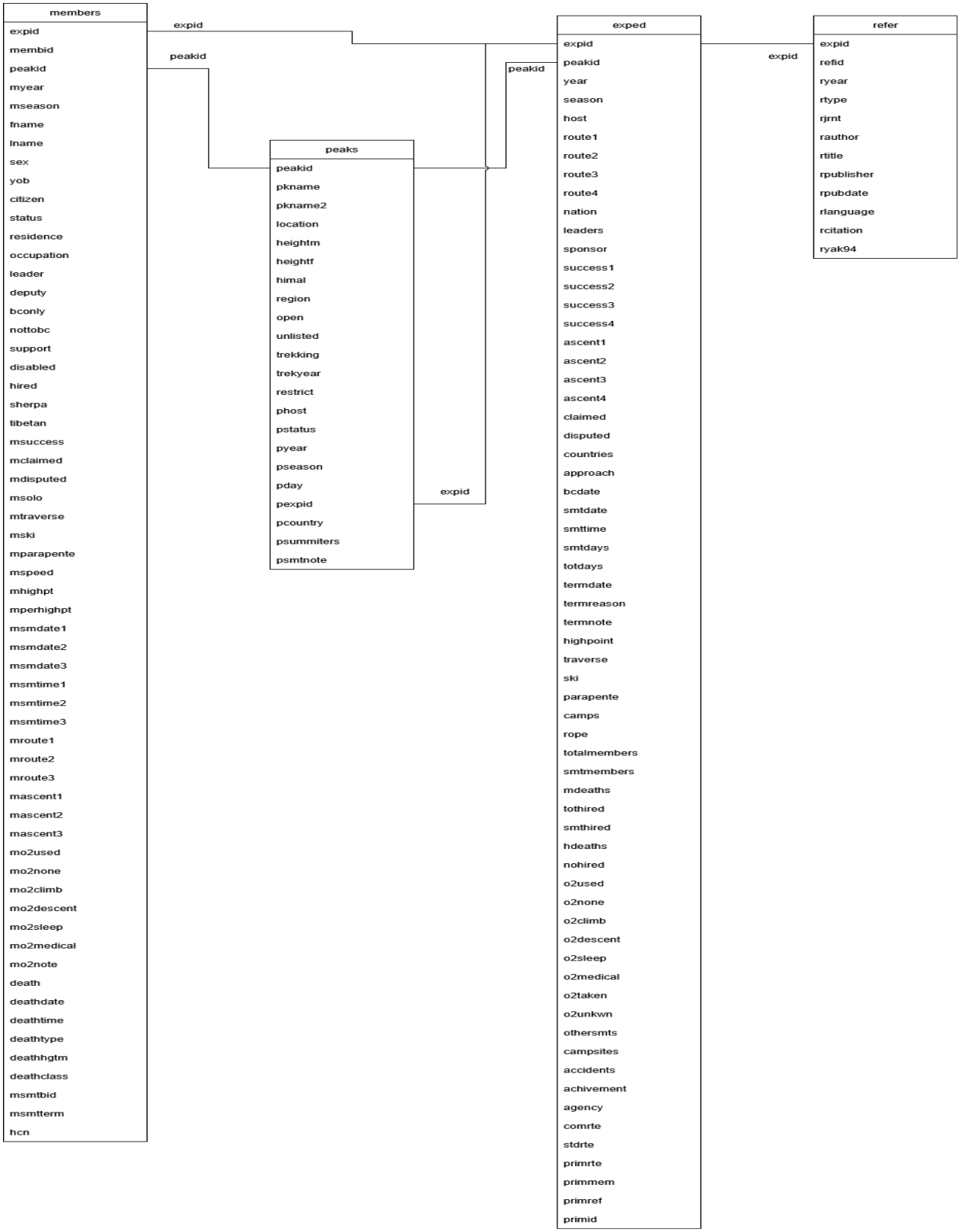
U ovome dijelu se prolazi kroz analizu samog izvora podataka kao i *OLTP* šeme. Dok će u drugom dijelu biti opis i analiza *OLAP* šeme samim tim i dimenzionih tabela i činjenične tabele.

4.1 Specifikacija izvora podataka

Izvor podatka koji je korišćen za izradu ovog projektnog zadatka jeste javno dostupni *Himalayan Expeditions* dataset sa *Keggle*. Svi fajlovi u ovom *dataset*-u su u obliku *CSV* fajlova. Svaki od fajlova sadrži veliki broj kolona i redova što čini ovaj *dataset* velikim i jako detaljnim. U nastavku slijede *CSV* fajlovi i njihova uloga u ovom *dataset*-u:

- → *exped.csv* - Sadrži podatke o konkretnim ekspedicijama: *expid (NK), peakid, year, host, route, nation, success, bcdate, smtdate, termreason, termdate, highpoint, o2used*... Ovo su neka od najbitnijih obilježja koja su bitna za skaldište podataka na osnovu korisničkih zahtjeva, ali i pored ovih obilježja ovaj fajl sadrži mnoštvo drugih obilježja što važi i za ostale fajlove, ali su birana samo ona obilježja koja su bitna za ovaj analitički sistem.
- → *members.csv* - Sadrži sve neophodne podatke za učešće pojedinačnih članova na određenim ekspedicijama: *expid i membid (NK), peakid, myear, yob, mseason, fname, lname, citizen status, residence, occupation, leader, sherpa, mhighpt, mperhighpt, msmdate1, msuccess, mo2used, death, msmterm*.. Naravno kao i kod *exped.csv* fajla izdvojena su kasnije samo ona obilježja koja su nam značajna za našu zvjezdastu šemu. Ono što je glavna problematika ovog *dataset*-a jeste napraviti tačnu razliku između *members.csv* fajla i *exped.csv* fajla. Važno za razumjeti je bilo da ovaj fajl predstavlja učešće jednog člana u jednoj ekspediciji i jedinstveno je određen sa *expid*, što predstavlja *id* ekspedicije na kojoj taj član učestvuje, i sa *membid* koji predstavlja *id* tog člana u toj ekspediciji. To znači da ovdje imamo i više istih članova koji se pojavljuju više puta ali u različitim ekspedicijama. *Peakid* predstavlja *id* vrha na koji se zaputio taj član tokom te ekspedicije. Bitno je takođe praviti razliku između obilježja u *exped.csv* i *members.csv*, gdje se obilježja u *members.csv* odnose isključivo samo na tog člana dok je prisustvovao u datoj ekspediciji. Odnose se samo na njega kao što je *mo2used* obilježje koje pokazuje da li je taj član korsitio dodatni kiseonik tokom ekspedicije, dok *o2used* u *exped.csv* predstavlja da li je barem jedan član te ekspedicije iskoristio kiseonik što ne mora da znači da je određeni pojedinac koristio kiseonik. Isto tako i za *success*, ekspedicija je uspješna ukoliko je barem neko došao do ciljnog vrha isto važi za najveću visnu koju su članovi dostigli tokom ekspedicije što ne znači da je pojedinac dostigao tu visinu jer je mogao odustati prije toga od ekspedicije i da se vrati nazad u kamp. Isto važi i za korišćenje kiseonika. Pravljenje razlike između ovih obilježja i fajlova je ključno za dalji nastavak.
- → *peaks.csv* - Sadrži podatke o vrhovima koji predstavljaju ciljne vrhove ekspedicija: *peakid (NK), pkname, location, heightm, region, open*... Navedena su neka od bitnih obilježja za dalju analizu.
- → *refer.csv* - Sadrži podatke o referencama odnosno o člancima pisanim za određene ekspedicije: *expid i refid (NK), ryear, rtype, rauthor*... Takođe možemo imati više istih referenci odnosno članaka koje govore o različitim ekspedicijama pa je zato *natural key* kombinacija *expid i refid*.

Pored ovih fajlova postoji još i *himalayan_data_dictonary.csv* fajl u kome se nalaze opisi svih kolona u ostalim fajlovima što je jako korisno za razumjevanje ovako detaljnog *dataset*-a i pogotovo jer su nazivi kolona u formi skraćenica. Ovaj fajl je jako koristan za detaljno i duboko razumjevanje *dataset*-a i pravljenje razlike između sličnih ali ipak različitih obilježja u različitim fajlovima. Na osnovu ovih *CSV* fajlova i njihovih opisa kreiran je sledeći *OLTP* model sa svim obilježjima i vezama putem stranih ključeva:


members
expid
membid
peakid
myear
mseason
fname
lname
sex
yob
citizen
status
residence
occupation
leader
deputy
bconly
nottobc
support
disabled
hired
sherpa
tibetan
msuccess
mclaimed
mdisputed
msolo
mtraverse
mski
mparapente
mspeed
mhighpt
mperhighpt
msmdate1
msmdate2
msmdate3
msmtime1
msmtime2
msmtime3
mroute1
mroute2
mroute3
mascent1
mascent2
mascent3
mo2used
mo2none
mo2climb
mo2descent
mo2sleep
mo2medical
mo2note
death
deathdate
deathtime
deathtype
deathhgtm
deathclass
msmtbid
msmtterm
hcn
expid
peakid
peaks
peakid
pkname
pkname2
location
heightm
heightf
himal
region
open
unlisted
trekking
trekyear
restrict
phost
pstatus
pyear
pseason
pday
pexpid
pcountry
psummiters
psmtnote
peakid
expid
exped
expid
peakid
year
season
host
route1
route2
route3
route4
nation
leaders
sponsor
success1
success2
success3
success4
ascent1
ascent2
ascent3
ascent4
claimed
disputed
countries
approach
bcdate
smtdate
smttime
smtdays
totdays
termdate
termreason
termnote
highpoint
traverse
ski
parapente
camps
rope
totalmembers
smtmembers
mdeaths
tothired
smthired
hdeaths
nohired
o2used
o2none
o2climb
o2descent
o2sleep
o2medical
o2taken
o2unkwn
othersmts
campsites
accidents
achivement
agency
comrte
stdrte
primrte
primmem
primref
primid
expid
refer
expid
refid
ryear
rtype
rjrnt
rauthor
rtitle
rpublisher
rpubdate
rlanguage
rcitation
ryak94

1 – OLTP šema

4.2 Specifikacija ciljanog Data Warehouse sistema

Krajnji sistem je projektovan u obliku zvjezdaste šeme sa jednom centralnom tabelom odnosno činjeničnom tabelom sa jasno definisanom granularnošću gdje jedan red predstavlja učestvovanje jednog pojedinca u određenoj ekspediciji. Činjenična tabela sadrži strane ključeve kojima je povezana sa ostalim dimenzionim tabelama.

4.2.1 Specifikacija zahtjevanih dimenzija

Dimenzione tabele predstavljaju deskriptivne tabele koje sadrže atribute koji nam pomažu u sagledavanju činjenične tabele na više različitih načina. Glavni cilj im je da dodaju dodatne deskriptivne opise mjerama iz činjenične tabele i da ih dodato prošire dodatnim kontekstom. Ključne su za filtriranje, poređenje, grupisanje i analizu svih podataka. U ovom modelu imamo sledeće dimenzione tabele:

- → *DimDate* – ova dimenzija sadrži sve neophodne vremenske atribute za vremenske analize (puni datum, dan, mjesec, kvartal i godina).
- → *DimNation* – dimenzija koja čuva sve podatke o državljanstvima svih učesnika, odnosno nacije kojoj pripadaju, takođe je i moguće da neko ima i dvojno državljanstvo što će takođe biti obrađeno (naziv nacije). Ova dimenzija je značajna za analizu performansi i uspješnosti članova ekspedicija na osnovu njihove nacionalne pripadnosti.
- → *DimRoute* – sve osnovne i javne rute koje se koriste za ekspedicije na ciljne vrhove će biti zabilježeni u ovoj dimenziji (naziv rute). Ova dimenzija je značajna za analzu performansi i kvaliteta ekspedicija i ruta prilikom biranja date rute za polazak na ekspediciju.
- → *DimStatus* – svaki od učesnika na nekoj ekspediciji ima neki status odnosno ulogu tokom te ekspedicije bilo to da li je običan planinar, lider, geograf, istoričar, kamerman ili nešto drugo (naziv statusa). U ovom *dataset*-u postoji veliki broj podgrupa odnosno podkategorija osnovnih uloga (npr. imamo pomoćnik koji nosi dodatni kiseonik i pomoćnik koji nosi užad neophodnu za planinarenje), i u *ETL* procesu je izvršen određeni nivo generalizacije ovih uloga odnosno statusa (npr. te pomoćnike možemo posmatrati pod jednim statusom Pomoćnik) radi lakše kasnije analize. Ova dimenzija je značajna za analizu performansi i uticaja učesnika sa određenim statusom na dalji tok ekspedicije.
- → *DimTerminationReason* – za svakog učesnika u ekspediciji je takođe zabilježen i razlog zašto je on u nekom momentu napustio ekspediciju i vratio se u početni kamp (naziv razloga za odustajanje). Postoji dosta razloga u polaznom *dataset*-u i ova dimenzija će zbog velikog broja različitih razloga biti od velikog značaja za analizu uzroka i poslijedica zašto je neki član napustio ekspediciju.
- → *DimPeak* – u ovoj dimenziji se nalaze svi neophodni podaci za sve ciljne vrhove na Himalajima (naziv vrha, lokacija na Himalajima, visina u metrima, Himalajski vijenac kome taj vrh pripada, region u kome se nalazi, oznaka da li je trajno otvoren za komercijalne ekspedicije ili ne). Ovdje se pored deskriptivnih obilježja korsite i jedno konstantno brojčano obilježje, odnosno ukupna visina tog planisnkog vrha, kao i oznaka da li je trajno otvoren ili zatvoren za komercijalne ekspedicije od strane vlade države kojoj taj vrh pripada. Ova dimenzij je značajna za kasnije analize uspješnosti ekspedicija na svim vrhovima kao i za analizu rizika i opasnosti koje ti vrhovi nose sa sobom.

→ *DimExpedition* – svi neophodni podaci vezani za konkretne ekspedicije će se nalaziti u ovoj dimenziji (država domaćin, sezona kada se ekspedicija održala, naziv ekspedicije, većinska nacija na toj ekspediciji, da li je ta ekspedicija došla do ciljnog vrha ili ne, najveća visina do koje su članovi ekspedicije došli, razlog zašto je većina članova te ekspedicije napustila taj pohod i vratila se u početni kamp). Sve zabilježene ekspedicije su se desile u prošlosti i za njih su zabilježeni svi neophodni i numerički i nenumerički podaci koji se neće mijenjati odnosno netekstualni podaci su konstanti pa iz tog razloga ovi atributi mogu da se smatraju kao obilježja ove dimenzije. Značajna je tabela za kasniju analizu performansi i uspješnosti svih učesnika na pojedinačnim ekspedicijama.
→ *DimMember* – svi izdvojeni jedinstveni i pojedinačni članovi ekspedicija su smješteni u ovoj dimenziji (ime, prezime, pol, godina rođenja, mjesto rođenja, zanimanje, oznaka da li je *Šerpas* ili ne). Ova dimenziona tabela je jako bitna i značajna za dalje sprovođenje svih analiza jer se većina tih analiza sprovodi nad nekim učesnicima.

4.2.2 Specifikacija zahtjeva mjera

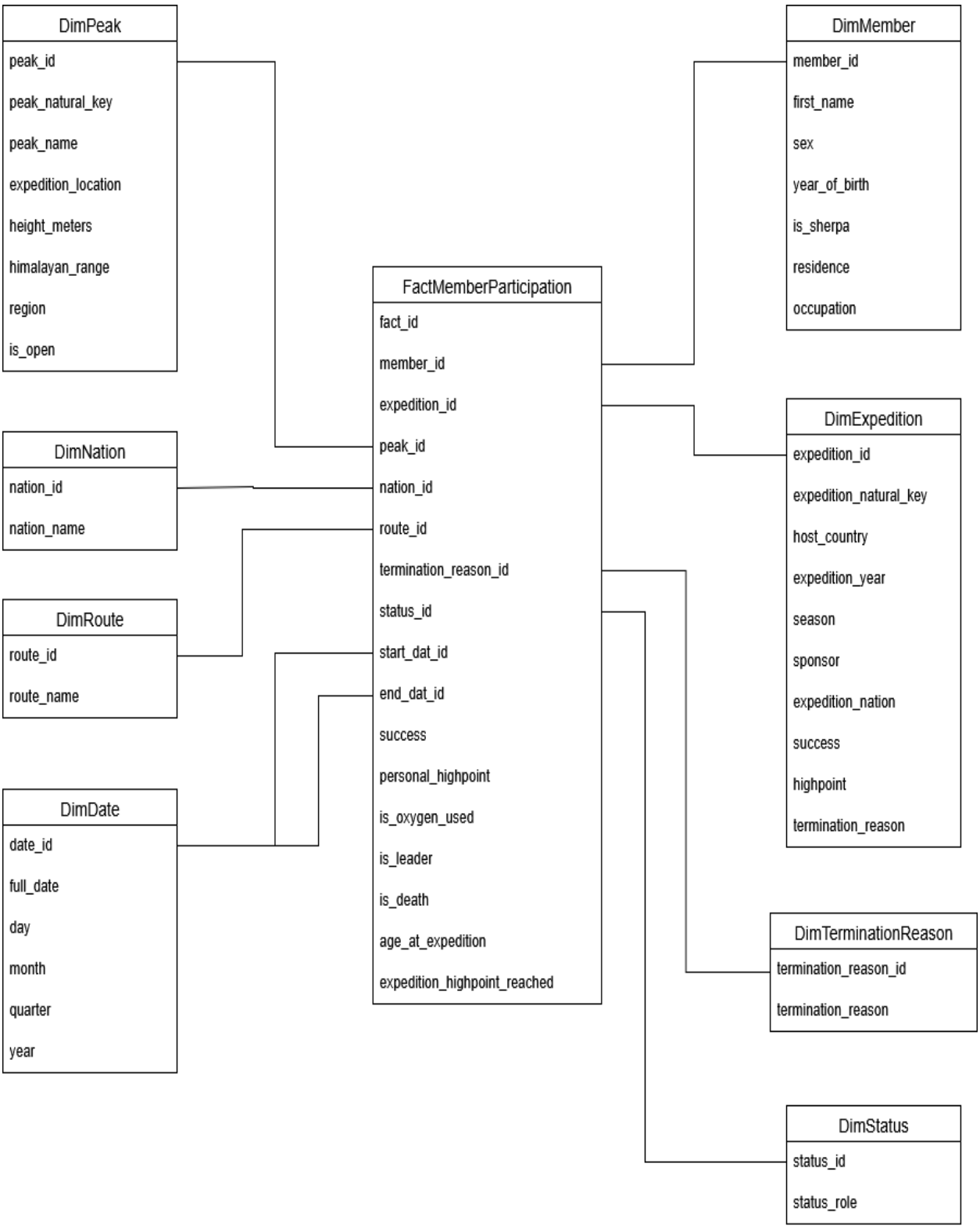
Mjere su kvantitivne vrijednosti činjeničnih tabela koje se analiziraju na osnovu dimenzionih tabela. Na osnovu svih definisanih mjera se generišu izvještaji i pružaju odgovori na korisnička pitanja i zahtjeve. Svaka činjenična taebla je uvijek vezana za određeni nivo granularnosti što je u ovom slučaju učestvovanje pojedinacnog člana u određenoj ekspediciji. U činjeničnoj tabeli definisane *OLAP* šeme se nalaze sledeće metrike koje će se poslije analizirati pomoću gore definisanih dimenzionih tabela:

→ *personal_highpoint* – predstavlja najveću visinu koju je taj član dostigao u svojoj ekspediciji.
→ *age_at_expedition* – broj godina koje je taj učesnik imao tokom svoje ekspedicije.

Pored ovih numeričkih mjera imamo takođe i logičke indikatore koji će posle biti najvažniji za dalju analizu:

→ *success* – oznaka da li je učesnik uspio sa da se popenje na ciljni vrh svoje ekspedicije.
→ *is_oxygen_used* – oznaka da li je taj učesnik koristio dodatni kisenoik tokom ekspedicije.
→ *is_leader* – oznaka da li je taj učesnik bio lider na svojoj ekspediciji.
→ *is_death* –oznaka da li je taj učesnik preživio svoju ekspediciju.
→ *expedition_highpoint_reached* – oznaka da li je taj učesnik dostigao najvišu tačku kao i njegova ekspedicija ili ne.

Data činjenična tabela je sa ostalim dimenzionim tabelama povezana putem stranih ključeva.


DimPeak
peak_id
peak_natural_key
peak_name
expedition_location
height_meters
himalayan_range
region
is_open
DimMember
member_id
first_name
sex
year_of_birth
is_sherpa
residence
occupation
FactMemberParticipation
fact_id
member_id
expedition_id
peak_id
nation_id
route_id
termination_reason_id
status_id
start_dat_id
end_dat_id
success
personal_highpoint
is_oxygen_used
is_leader
is_death
age_at_expedition
expedition_highpoint_reached
DimExpedition
expedition_id
expedition_natural_key
host_country
expedition_year
season
sponsor
expedition_nation
success
highpoint
termination_reason
DimNation
nation_id
nation_name
DimRoute
route_id
route_name
DimDate
date_id
full_date
day
month
quarter
year
DimTerminationReason
termination_reason_id
termination_reason
DimStatus
status_id
status_role

2 – OLAP šema

5. Opis ETL procesa

Čitav *ETL* proces započinje sa kreiranjem *Execute SQL Task*-a u kome ćemo izvršavati uslovni *drop* činjenične tabele, dimenzionih tabela i šeme redom kako bi smo omogućili kreiranje tabela u narednom koraku:

```
IF OBJECT_ID('Expeditions.FactMemberParticipation', 'U') IS NOT NULL
DROP TABLE Expeditions.FactMemberParticipation

IF OBJECT_ID('Expeditions.DimDate', 'U') IS NOT NULL
DROP TABLE Expeditions.DimDate

IF OBJECT_ID('Expeditions.DimPeak', 'U') IS NOT NULL
DROP TABLE Expeditions.DimPeak

IF OBJECT_ID('Expeditions.DimExpedition', 'U') IS NOT NULL
DROP TABLE Expeditions.DimExpedition

IF OBJECT_ID('Expeditions.DimNation', 'U') IS NOT NULL
DROP TABLE Expeditions.DimNation

IF OBJECT_ID('Expeditions.DimRoute', 'U') IS NOT NULL
DROP TABLE Expeditions.DimRoute

IF OBJECT_ID('Expeditions.DimMember', 'U') IS NOT NULL
DROP TABLE Expeditions.DimMember

IF OBJECT_ID('Expeditions.DimTerminationReason', 'U') IS NOT NULL
DROP TABLE Expeditions.DimTerminationReason

IF OBJECT_ID('Expeditions.DimStatus', 'U') IS NOT NULL
DROP TABLE Expeditions.DimStatus

IF SCHEMA_ID('Expeditions') IS NOT NULL
DROP SCHEMA Expeditions
```

3 – Drop Tables

U daljem *SSIS Flow*-u imamo sljedeći *Execute SQL Task* koji će nam služiti za kreiranje šeme, dimenzionih tabela i činjenične tabele redom:

```sql
CREATE SCHEMA Expeditions

CREATE TABLE Expeditions.DimDate(
    date_id int not null identity,
    full_date date not null,
    day nvarchar(15) not null,
    month nvarchar(15) not null,
    quarter int not null,
    year int not null

    constraint PK_DimDate primary key(date_id)
)

CREATE TABLE Expeditions.DimPeak (
    peak_id int not null identity,
    peak_natural_key nvarchar(50) not null,
    peak_name nvarchar(100) not null,
    expedition_location nvarchar(500) not null,
    height_meters int not null,
    himalayan_range nvarchar(100) not null,
    region nvarchar(100) not null,
    is_open bit not null

    constraint PK_DimPeak primary key(peak_id)
)

CREATE TABLE Expeditions.DimNation (
    nation_id int not null identity,
    nation_name nvarchar(50) not null

    constraint PK_DimNation primary key(nation_id)
)
```

4 – Create Tables(1)

```sql
CREATE TABLE Expeditions.DimExpedition (
    expedition_id int not null identity,
    expedition_natural_key nvarchar(20) not null,
    host_country nvarchar(30) not null,
    expedition_year int not null,
    season nvarchar(20) not null,
    sponsor nvarchar(100) not null,
    expedition_nation nvarchar(50) not null,
    success bit not null,
    highpoint int not null,
    termination_reason nvarchar(150) not null

    constraint PK_DimExpedition primary key(expedition_id)
)

CREATE TABLE Expeditions.DimRoute (
    route_id int not null identity,
    route_name nvarchar(70) not null

    constraint PK_DimRoute primary key(route_id)
)

CREATE TABLE Expeditions.DimMember (
    member_id int not null identity,
    first_name nvarchar(50) not null,
    last_name nvarchar(50) not null,
    sex char(1) not null,
    year_of_birth int not null,
    is_sherpa bit not null,
    residence nvarchar(255) not null,
    occupation nvarchar(255) not null

    constraint PK_DimMember primary key(member_id)
)
```

5 – Create Tables(2)

```sql
CREATE TABLE Expeditions.DimTerminationReason (
    termination_reason_id int not null identity,
    termination_reason nvarchar(150) not null

    constraint PK_DimTerminationReason primary key(termination_reason_id)
)

CREATE TABLE Expeditions.DimStatus (
    status_id int not null identity,
    status_role nvarchar(100) not null

    constraint PK_DimStatus primary key(status_id)
)
```

6 – Create Tables(3)

```sql
CREATE TABLE Expeditions.FactMemberParticipation (
    fact_id int not null identity,
    member_id int not null,
    expedition_id int not null,
    peak_id int not null,
    nation_id int not null,
    route_id int not null,
    termination_reason_id int not null,
    status_id int not null,
    start_dat_id int not null,
    end_dat_id int not null,

    success bit not null,
    personal_highpoint int not null,
    is_oxygen_used bit not null,
    is_leader bit not null,
    is_death bit not null,
    age_at_expedition int not null,
    expedition_highpoint_reached bit not null

    constraint PK_Fact primary key(fact_id),
    constraint FK_Fact_Member foreign key(member_id)
        references Expeditions.DimMember(member_id),
    constraint FK_Fact_Expedition foreign key(expedition_id)
        references Expeditions.DimExpedition(expedition_id),
    constraint FK_Fact_Peak foreign key(peak_id)
        references Expeditions.DimPeak(peak_id),
    constraint FK_Fact_Nation foreign key(nation_id)
        references Expeditions.DimNation(nation_id),
    constraint FK_Fact_Route foreign key(route_id)
        references Expeditions.DimRoute(route_id),
    constraint FK_Fact_TerminationReason foreign key(termination_reason_id)
        references Expeditions.DimTerminationReason(termination_reason_id),
    constraint FK_Fact_Status foreign key(status_id)
        references Expeditions.DimStatus(status_id),
    constraint FK_Fact_Start foreign key(start_dat_id)
        references Expeditions.DimDate(date_id),
    constraint FK_Fact_End foreign key(end_dat_id)
        references Expeditions.DimDate(date_id)
)
```

7 – Create Fact Table

Za potrebe daljeg rada prilikom popunjavanja činjenične tabele bilo je potrebno dodati redove u neke kreirane dimenzije koje će reprezentovati nedostajuće vrijednosti ukoliko ima takvih u činjeničnoj tabeli:

```
SET IDENTITY_INSERT Expeditions.DimRoute ON;

INSERT INTO Expeditions.DimRoute(route_id, route_name)
VALUES (-1, 'Unknown');

SET IDENTITY_INSERT Expeditions.DimRoute OFF;

SET IDENTITY_INSERT Expeditions.DimNation ON;

INSERT INTO Expeditions.DimNation (nation_id, nation_name)
VALUES (-1, 'Unknown');

SET IDENTITY_INSERT Expeditions.DimNation OFF;
```

8 – Insert Unkonwn data

5.1 Punjenje dimenzionih tabela

Nakon kreiranja neophodne šeme, dimenzionih tabela i činjenične tabele neophodno je popuniti te tabele sa izvornim podacima odnosno sa podacima koje ćemo dobijati iz izvornih *CSV* fajlova. Što se tiče punjenja dimenzionih tabela prvo je popunjena vremenska dimenzija i to na način da je kreiran novi *Execute SQL Task* koji će izvršavati sledeći *SQL Query* u cilju punjenja svih kolona u vremenoskoj dimenziji od 1900 do 2026 godine.

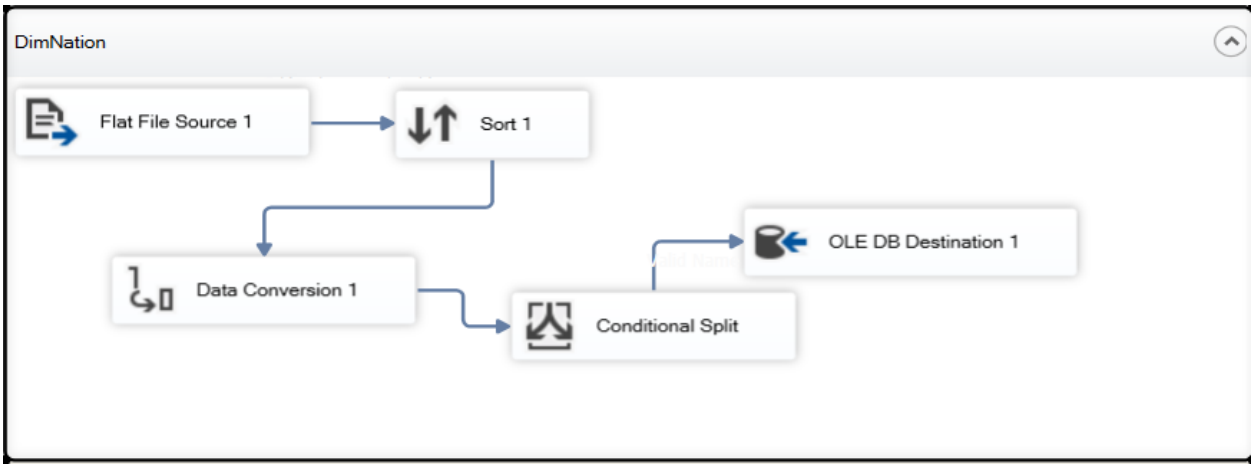
```
SET IDENTITY_INSERT Expeditions.DimDate ON;

INSERT INTO Expeditions.DimDate (date_id, full_date, day, month, quarter, year)
VALUES (-1, '1899-01-01', 'Unknown', 'Unknown', 0, 0);

SET IDENTITY_INSERT Expeditions.DimDate OFF;
GO

DECLARE @StartDate DATE = '1900-01-01';
DECLARE @EndDate   DATE = '2026-12-31';

WHILE @StartDate <= @EndDate
BEGIN
    INSERT INTO Expeditions.DimDate (full_date, day, month, quarter, year)
    VALUES (
        @StartDate,
        DATENAME(DW, @StartDate),
        DATENAME(MONTH, @StartDate),
        DATEPART(QUARTER, @StartDate),
        YEAR(@StartDate)
    );

    SET @StartDate = DATEADD(DAY, 1, @StartDate);
END;
GO
```

9 – Insert DimDate

Nakon popunjavanja vremenske dimenzije u posebnom *Execute SQL Task*-u neophodno je popuniti i ostale vremenske dimenzije i zbog toga se koristi novi *Data Flow Task* u okviru koga su grupisani svi neophodni *Data Task*-ovi za ekstrakciju, transformaciju i učitavanje podataka iz izvornih *CSV* fajlova u određene ciljne dimenzione tabele.

DimNation

Za punjenje dimenzione tabele *DimNation*, koja će da reprezentuje kojoj naciji pripada određeni član ekspedicije, kao izvor korišćen je *Flat File Source* jer je izvor naših podataka koje ćemo koristiti za punjenje date tabele zapravo *CSV* fajl. Kao konkretan izvor korišćen je *members.csv* fajl u kome se nalaze podaci o svim učešćima članova na svim ekspedicijama. Iz tog fajla je korišćeno za potrebe punjenja ove tabele samo *citizen* obilježje koje označava kojoj nacionalnosti pripada dati učesnik. Pošto u *members.csv* fajlu imamo učešća svih članova moramo izdvojiti pojedinačne nacije jer nam se može desiti da dva različita učesnika pripadaju istoj naciji, u tom slučaju bi smo imali duplirane zapise u tabeli, i zato se korsisti *Sort* komponenta kako bi smo sortirali po obilježju *citizen* i kako bi smo izbacili duplikate, što je jedna od opcija *Sort* komponente. Nakon toga bilo je neophodno sa komponentom *Data Conversion* da se obilježje *citizen* konvertuje u *unicode WSTR*, kako bi smo usaglasili tipove podataka sa tipovima podataka u bazi. Nakon toga se u komponenti *Conditional Split* izbacuju one nacije koje su nepoznate odnosno prazan *string* ili *null* vrijednosti, dok smo valide vrijednosti proslijedili dalje u *OLE DB Destination* komponentu, zato što rezultate učitavanja i transformacije podataka iz izvora moramo da učitamo u *SQL* bazu podataka. U okviru te komponente mapirana je konvertovana kolona *citizen* sa obilježjem *nation_name* u bazi podataka. Takođe je moguće da neki učesnici imaju i dvojna državljanstva što se računa kao posebna nacionalnost u tabeli.

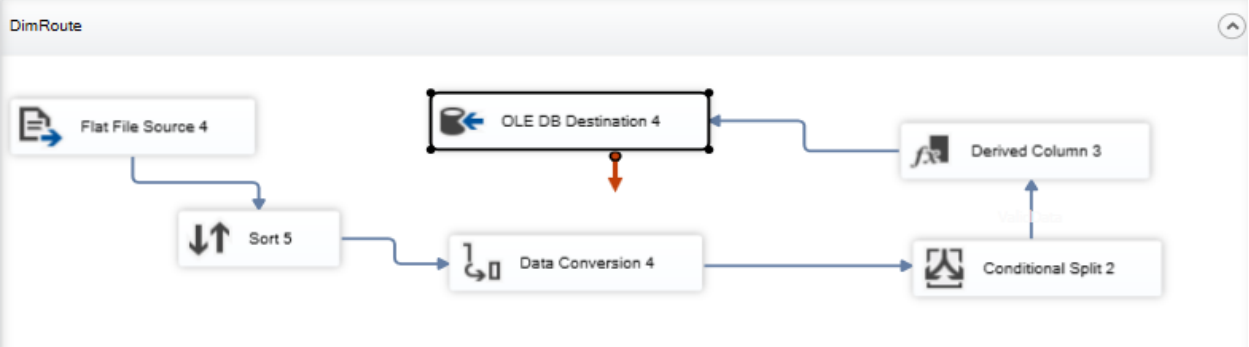


10 – DimNation

DimRoute

Sledeća dimenziona tabela koju je neophodno napuiniti je *DimRoute*, koja će da skladišti podatke o rutama kojima su se kretali učesnici ekspedicija na svoje ciljne vrhove. Kao izvor podataka korišćen je *Flat File Source* jer se podatak koji je neophodan za punjenje ove dimenzione tabele nalazi u *exped.csv* fajlu a to je *route1* obilježje. Nakon toga slično kao za prošlu dimenziju koristi se komponenta *Sort* da izbacimo duplikate jer je moguće da se na više ekspedicija ide istom rutom, time izbjegavamo duplikate u dimenzionoj tabeli. Zatim je neophodno i konvertovati to

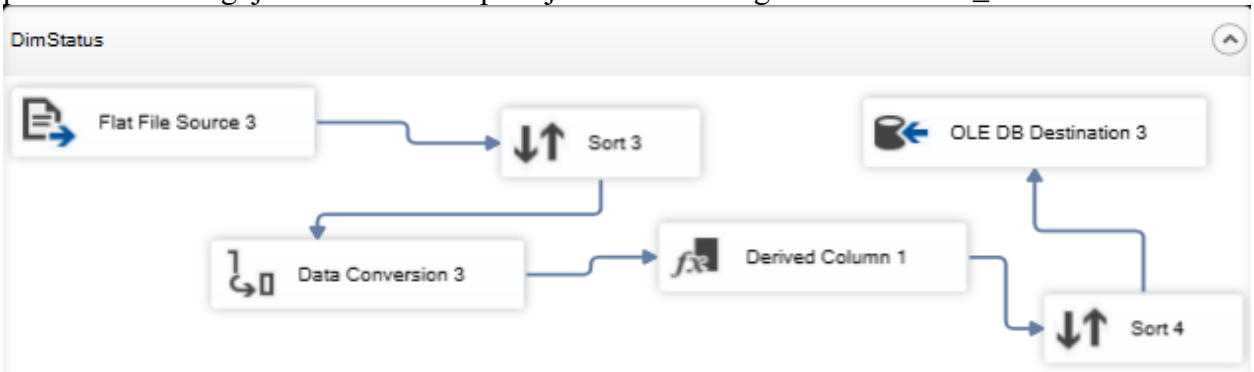
obilježje u *unicode WSTR* komponentom *Data Conversion*. Takođe postoje i polja koja su nepoznata ili *null* i takve redove izbacujemo komponentom *Conditional Split*. Komponentom *Derived Column* izbaucjemo nepotrebne razmake i karaktere u korišćenim rutama. Dobijeni podaci se šalju u *OLE DB Destination* za konačno upsisivanje u *DimRoute* dimenziju, pri čemu je izvršeno mapirane konvertovanog obilježja *route1* u *route_name* u tabeli.



11 – DimRoute

DimStatus

U činjeničnoj tabeli *DimStatus* će se nalaziti podaci o tome koje su statuse imali određeni učesnici u svojim ekspedicijama. U polaznom *dataset*-u imamo veliki broj uloga odnosno statusa koje je učesnik imao tokom svojih ekspedicija. Radi kasnije lakše analize i jednostavnosti odlučeno je da se sprovede određeni stepen generalizacije tih statusa (npr. *Support02* i *SupportRope* je predstavljeno kao *SUPPORT*). Kao izvor se opet korsiti *Flat File Source* jer obilježje *status* uzimamo iz *members.csv* fajla. Kao i u prethodnim dimenzijama koristimo komponentu *Sort* za sortiranje dobijenih podataka i izbacivanje duplikata, jer se u tom fajlu nalaze sva učešća članova ekspedicija i statusi koje su oni imali na tim ekspedicijama se često ponavljaju. Opet je bilo neophodno konvertovati *status* u *unicode WSTR* pomoću komponente *Data Conversion*. Nakon toga u komponenti *Derived Column* vršimo kompletnu logiku generalizacije statusa koji učesnici mogu imati u svojim ekspedicijama. Prilikom generalizovanja opet smo imali određeni mali broj duplikata pa smo i njih eliminisali korišćenjem komponente *Sort*. Takvi transformisani podaci se šalju komponenti *OLE DB Destination* kojom upisujemo podatke u bazu i gdje smo izvršili mapiranje konvertovanog statusa u *status_role* u bazi.

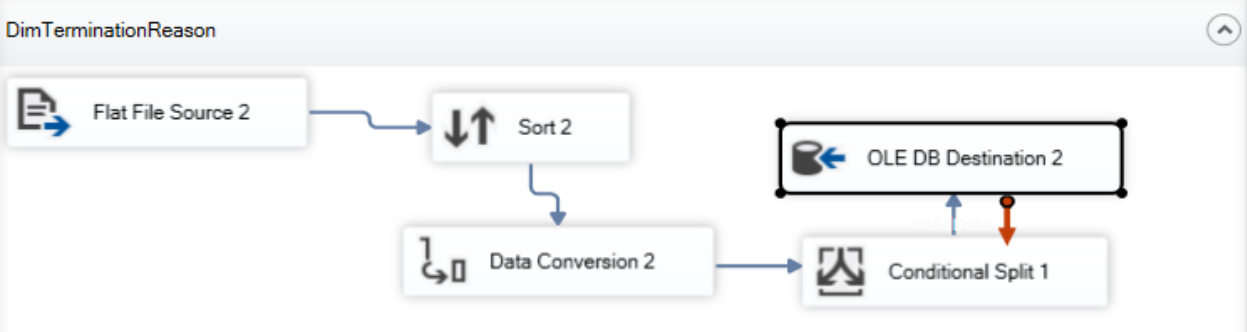


12 – DimStatus

DimTerminationReason

Svaki od učesnika na ekspedicijama ima neki svoj razlog zašto je završio tu ekspediciju bilo to da je dostizanje ciljnog vrha, ili je nepozanto, ili je nešto drugo kao npr. manjak kiseonika ili

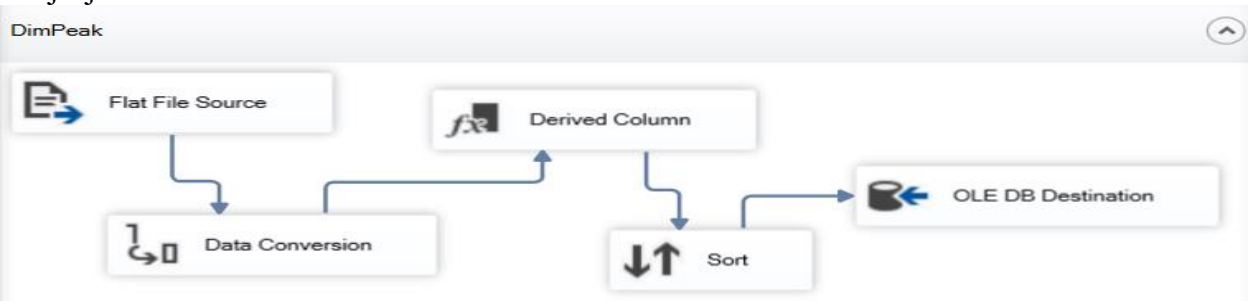
lavina ili bolest ili nešto drugo. Takve podatke ćemo čuvati u dimenzionoj tabeli *DimTerminationReason*. Kao izvor za punjenje ove činjenične tabele koristimo *Flat File Source* jer se obilježje koje nam je potrebno nalazi u fajlu *members.csv*, a to je *msmtterm*. Opet, da bi smo izbjegli duplikate koristimo *Sort* komponentu. Da bi smo uskladili sve tipove podataka neophodno je i konvertovati obilježje u *unicode WSTR* korišćenjem komponente *Data Conversion*. Polja koja su nepoznata, *null* ili imaju neki drugi zapis kako bi indikovala da nisu poznata, su izbačena komponentom *Conditional Split*. Dok se ispravni podaci šalju u *OLE DB Destination* za konačno upisivanje u bazu, tu je takođe izvršeno mapiranje konvertovanog obilježja sa *termination_reason* u bazi podataka.



13 – DimTerminationReason

DimPeak

Svi neophodni podaci o ciljnim vrhovima ekspedicija će se nalaziti u dimenzionoj tabeli *DimPeak*. Kao izvor podataka za ovu dimenzionu tabelu koristimo *peaks.csv* i zato je neophodana *Flat File Source* komponenta odakle uzimamo obilježja *peakid* koje ćemo koristiti kao *natural key* u ovoj dimenziji, zatim *pkname, location, heightm, himal, region, open*. Sve tekstuale podatke smo morali konvertovati u *unicode WSTR* korišćenjem komponente *Data Conversion*, a to su *pkname, location, himal, region i peakid*. Zatim u *Derived Column* komponenti su otklonjeni svi nepotrebni karakteri iz konvertovane lokacije tog vrha na Himalajima. Zatim su svi vrhovi sortirani po visni komponentom *Sort* ali ovaj put bez opcije uklanjanja duplikata jer se može desiti da dva ili više vrhova imaju istu visinu i time bi smo izgubili podatke o tom vrhu. Tako transformisani podaci se šalju *OLE DB Destination* komponenti za finalno upisivanje u bazu podataka, tu su takođe izvršena neophodna mapiranja sa obilježjima u bazi.

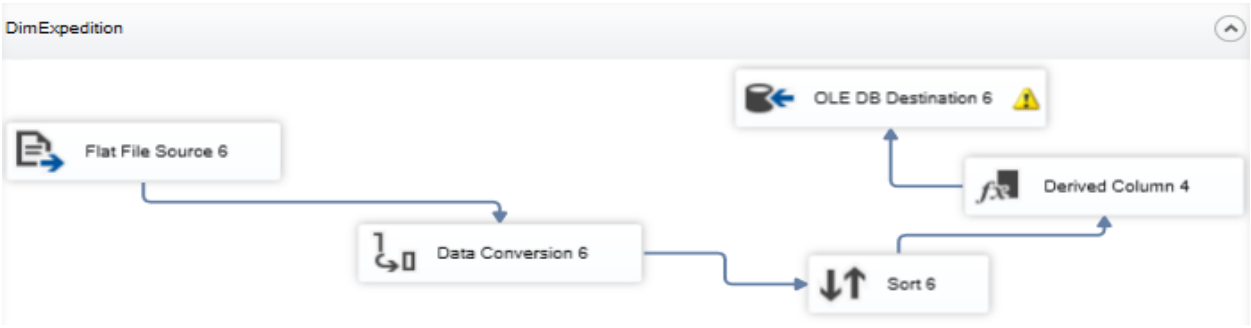


14 – DimPeak

DimExpedition

Sve neophodne podatke o zabilježenim ekspedicijama u periodu od 124 godine će se čuvati u dimenziji *DimExpedition*. Kao izvor za ove podatke se koristi fajl *exped.csv* i iz tog razloga je neophodne komponenta *Flat File Source*. U toj komponenti biramo obilježja iz *exped.csv* fajla

koja su nam neophodna a to su *expid* kojeg ćemo mapirati u *natural key* ekspedicije u ovoj dimenziji, zatim *year, season, host, nation, sponsor, success, termreason i highpoint*. Svi tekstualni podaci su konvertovani u *unicode WSTR* u komponenti *Date Conversion*. Ztim su svi podaci sortirani u komponenti *Sort* po godini kada se ta ekspedicija odvijala, ali ne korsistimo opciju za uklanjanje duplikata jer želimo da zadržimo sve ekspedicije. Sada u narednoj komponenti, *Derived Column* komponenti, radimo prečišćavanje svih podataka i za sve nepoznate ili prazne podatke stavljamo *Unknown*. Nakon svih ovih transformacija i prečišćavanja podataka sada upisujemo podatke u dimenzionu tabelu koristeći komponentu *OLE DB Destiantion*, u kojoj su takođe izvršena neophodna mapiranja odgovarajućih kolona.



15 – DimExpedition

DimMember

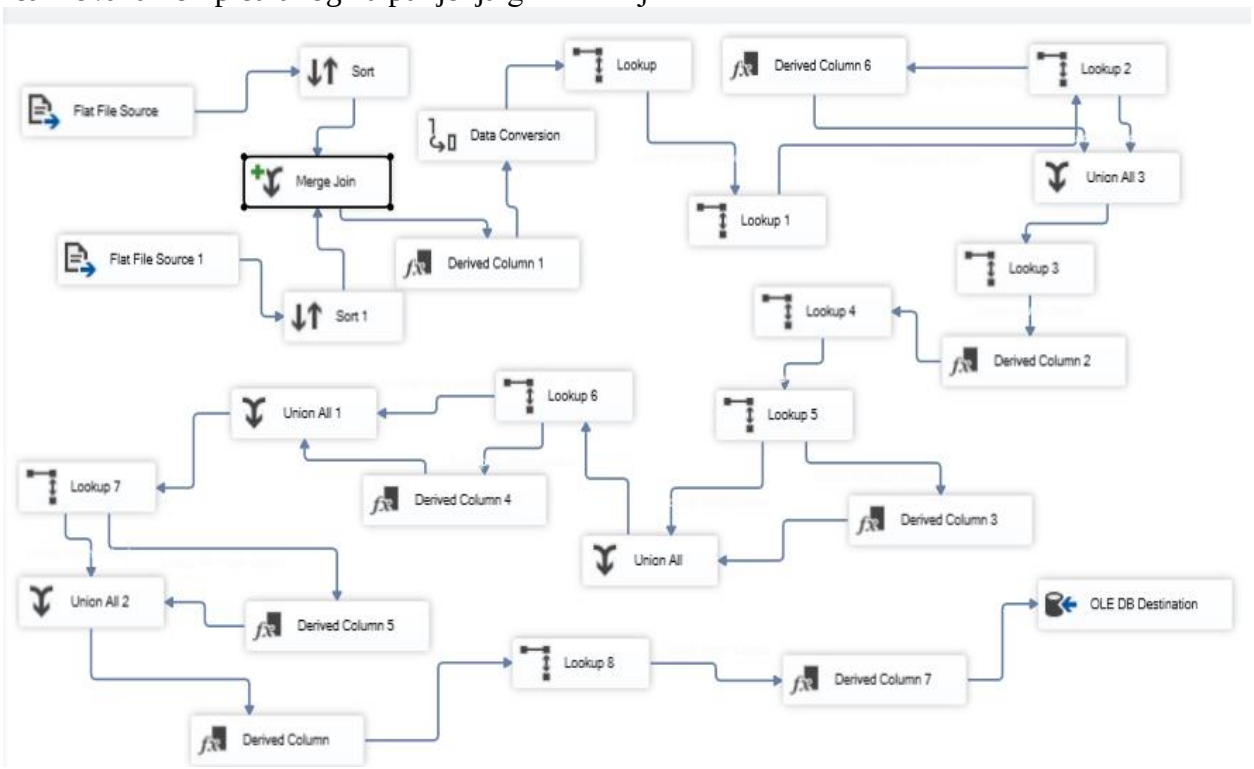
Sada slijedi jako važan korak da se izdvoje svi pojedinačni članovi ekspedicija iz *members.csv* fajla. Oni su u tom fajlu određeni sa *expid* i *membid*, ali taj *membid* samo predstavlja *id* tog učestnika u datoj ekspediciji. Može se desiti da se isti član pojavi drugi put sa istim *membid* ali u okviru druge ekspedicije, pa je neophodno napraviti razliku između toga i izdvojiti samo pojedinače učesnike sa njihovim ličnim podacima i smejstiti ih u dimenzionu tabelu *DimMember*. Pošto koristimo *members.csv* fajl kao izvor, opet se koristi *Flat File Source* kao izvor podataka. Od mnogobrojnih obilježja biće uzeti samo *fname, lname, sex, yob* odnosno godina rođenja, *residence, occupation i sherpa* koje predstavlja oznaku da li je taj član pripadnik *Šerpasa* ili ne. Zatim smo grupisali sve članove u komponenti *Aggregate* po svim ulaznim obilježjima kako bi smo izvukli tačno jedinstvene članove. Rađeno je po svim obilježjima iz razloga zato što se može desiti da je više članova rođeno iste godine i da ima isto ime i prazime, pa smo radi povećavanja kriterijuma za jedinstvene članove koristili grupisanje po svim obilježjima. Nakon toga se u komponenti *Data Conversion* sva tekstualna obilježja konvertuju u *unicode WSTR*. Potrebno je zatim pročistiti sve dobijene podatke i da se umjesto nepostojućih i praznih vrijednosti stavi *Unknown* vrijednost, što je urađeno u *Derived Column* komponenti. Pored toga godina rođenja za učesnike se postavlja na vrijednost –1 ukoliko nije poznata ili je prazna kako bi smo lakše posle u analitici i izvještajima postavili uslov u *WHERE* klauzuli da ignorišemo nepostojeće godine rođenja, a tada će to biti slučaj kada je godina rođenja veća od 0. Postoji i mogućnost da je neki učesnik promjenio svoje prebivalište ili svoje zanimanje tokom vremena pa da je u fajlu *members.csv* tokom više svojih ekspedicija upisivan sa drugačijim prebivalištem i zanimanjem. Pa je mogući slučaj da imamo istu osobu sa različitim zanimanjima ili prebivalištem. Taj problem je riješen sortiranjem svih na kraju dobijenih, konvertovanih i pročišćenih učesnika po svim drugim obilježjima osim za zanimanje i prebivalište. To je sve urađeno u komponenti *Sort* i u kojoj smo uklonili te duplikate ukoliko ih ima i na taj način zadržavajući poslednje mjesto prebivališta i zanimanje. Sve ove podatke šaljemo na *OLE DB*

Destination komponentu koja će upisati transforimisane podatke u dimenziju *DimMember*, Tu su takođe odrađena i mapiranja svih odgovarajućih obilježja.

16 – DimMember

5.2 Punjenje činjenične tabele

Nakon punjenja dimenzionih tabela, napravljen je novi *Data Flow Task* u okviru koga će biti realizovana kompletna logika punjenja glavne činjenične tabele.



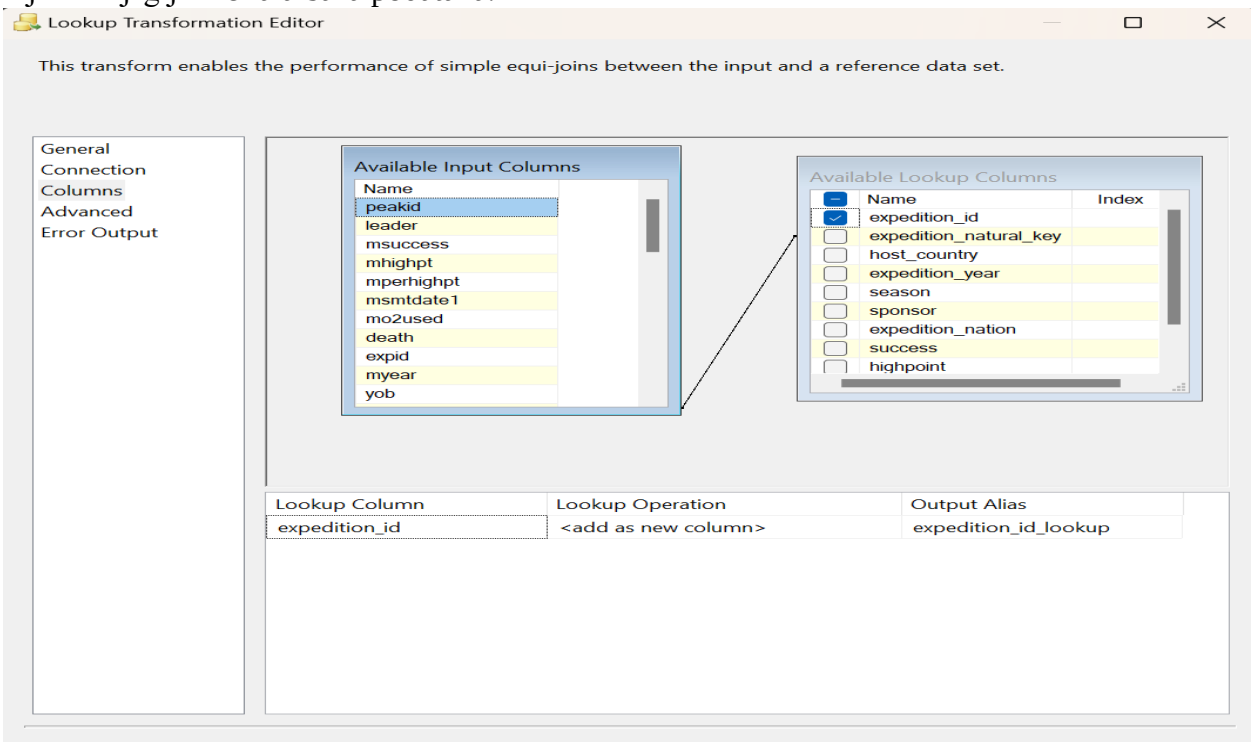
17 - FactTable

Izvorni fajlovi koji će se koristiti za punjenje činjenične tabele su members.csv i *exped.csv*, iz tog razloga sada se koriste dvije *Flat File Source* komponente u okviru kojih ćemo uzimati neophodna obilježja a to su: *expid, peakid, myear, fname, lname, sex, yob, citizen, status, residence, occupation, leader, sherpa, msuccess, mgighpt, mperhighpt, msmdate1, mo2used, death, msmtterm* iz *members.csv* fajla i *expid, route1, bcdate* iz *exped.csv* fajla. Ideja je da se podaci koje smo dobili iz ova dva fajla spoje putem zajedničkog ključa, što je u ovom slučaju

expid. Komponenta koja je idealna za povezivanje podataka iz dva fajla po zajedničkom ključu je *Merge Join*. Ali prije toga je neophodno sortirati oba toka podataka na isiti način po zajedničkom obilježju po kome će se povezivati. To je odrađeno u odvojenim *Sort* komponentama, koje zatim šalju podatke *Merge Join* komponenti za spajanje. Nakon spajanja vršimo pročišćavanje podataka i njihovu konverziju u odgovarajuće tipove podataka pomoću *Derived Column* i *Data Conversion* komponente.

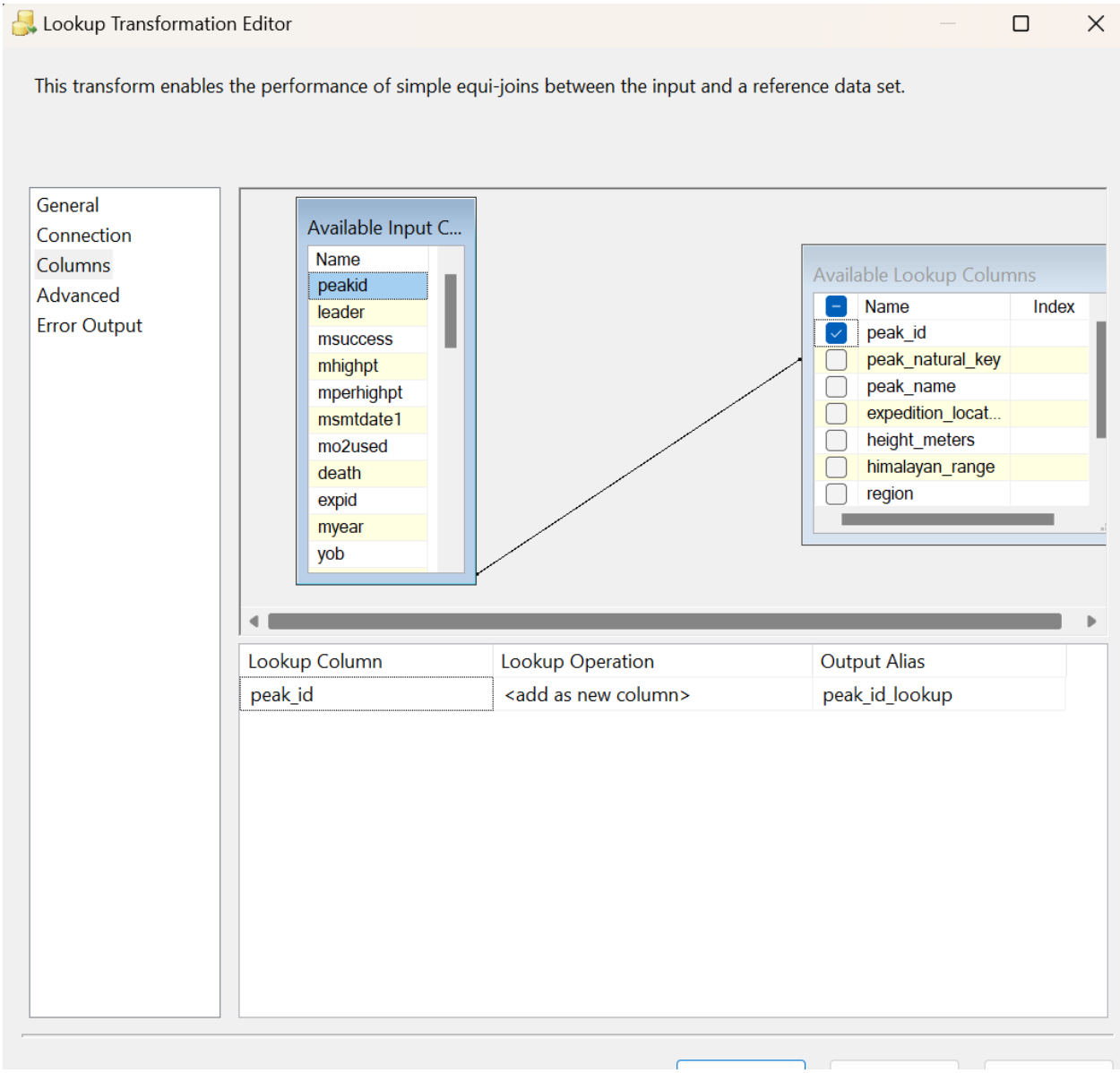
Za povezivanje činjenične tabele sa dimenzionim tabelama, odnosno za pronalaženje adekvatnih ključeva u bazi podataka za dimenzione tabele na osnovu datih obilježja korišćena je komponenta *Lookup*. Ta komponenta pretražuje jedinstvenu vrijednost nekog obilježja u tabeli baze podataka i vraća nam vrijednosti željenih obilježja za pronađeni red u bazi podataka, u većini slučajeva i u ovom slučaju uvijek će vraćati vrijednost primarnog ključa za pronađeni red. Takav ključ će se upisivati kao vrijednost stranog ključa u činjeničnoj tabeli. To je osnovna ideja kako funkcioniše ova komponenta i kako će se realizovati popunjavanje vrijednosti stranih ključeva u činjeničnoj tabeli.

Nakon transformacije i konvertovanja podataka sada slijedi prvi u nizu *Lookup* komponenti, koji će nam služiti za dobijanje vrijednosti stranog ključa *expedition_id* iz dimenzione tabele *DimExpedition*. U *Lookup* komponenti pretražujemo tabelu *DimExpedition* po vrijednosti *natural_key* ključa, odnosno ključa iz prvobitnog seta podataka i vraćamo vrijednost primarnog ključa kojeg je kreirala baza podataka.



18 – Lookup Expedition

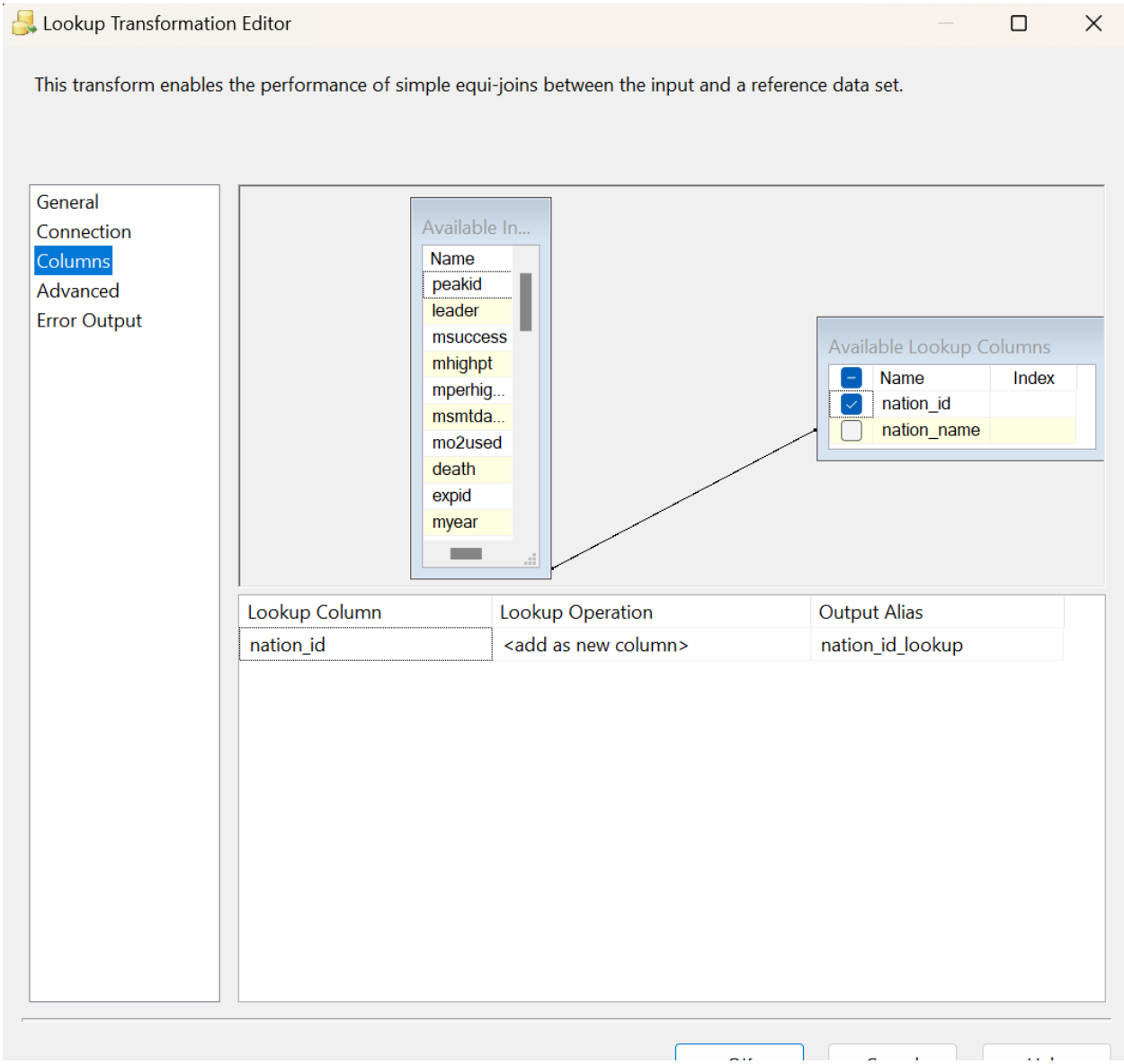
Naredna *Lookup* komponenta je jako slična prethodnoj jer će se opet pretraživati tabela na osnovu vrijednosti *natural_key*. Ovaj put se traži vrijednost *natural_key* za vrh na koji se učesnik ekspedicije sa *id*, koji je nađen u prethodnoj *Lookup* komponenti, uputio. Tu informaciju tražimo u dimenzionoj tabeli *DimPeak* i tu ćemo tražiti vrijednost obilježja *natural_key* na osnovu *peakid* iz izvornih podataka i vratiti primarni ključ za pronađenu torku. Na taj način dobijamo i podatke o ciljnom vrhu na koji se član ekspedicije zaputio.



19 – Lookup Peak

Sledeća *Lookup* komponenta nam služi za dobijanje vrijednosti stranog ključa *nation_id* iz dimenzione tabele *DimNation*. Ova informacija nam je bitna da znamo kojoj nacionalnosti pripada učesnik na ekspediciji. Ovog puta nemamo *natural_key* za povezivanje pa ćemo povezivanje vršiti na osnovu naziva nacije, odnosno *nation_name*. Prilikom punjenja dimenzione tabele *DimNation*, u tu tabeli su se unosile samo jedinstvene nacije bez duplikata i zbog toga će komponenta *Lookup* moći pronaći jedinstvenu vrijednost za *nation_name* i da time vrati odgovarajuću vrijednost primarnog ključa za tu naciju i ta vrijednost će biti upisana kao vrijednost stranog ključa *nation_id* u činjeničnoj tabeli. Takođe u spojenom izvoru podataka je moguće da nemamo upisanu vrijednost za naziv nacije i da je *Lookup* komponenta ne može pronaći u dimenzionoj tabeli *DimNation*. U tom slučaju *No Match Output* iz komponente *Lookup* šalejmo u *Derived Colum* komponentu koja će postaviti vrijednost tog stranog ključa na –1 i na taj način ukazujemo da je vrijednost tog stranog ključa nepoznata i da je ignorišemo prilikom

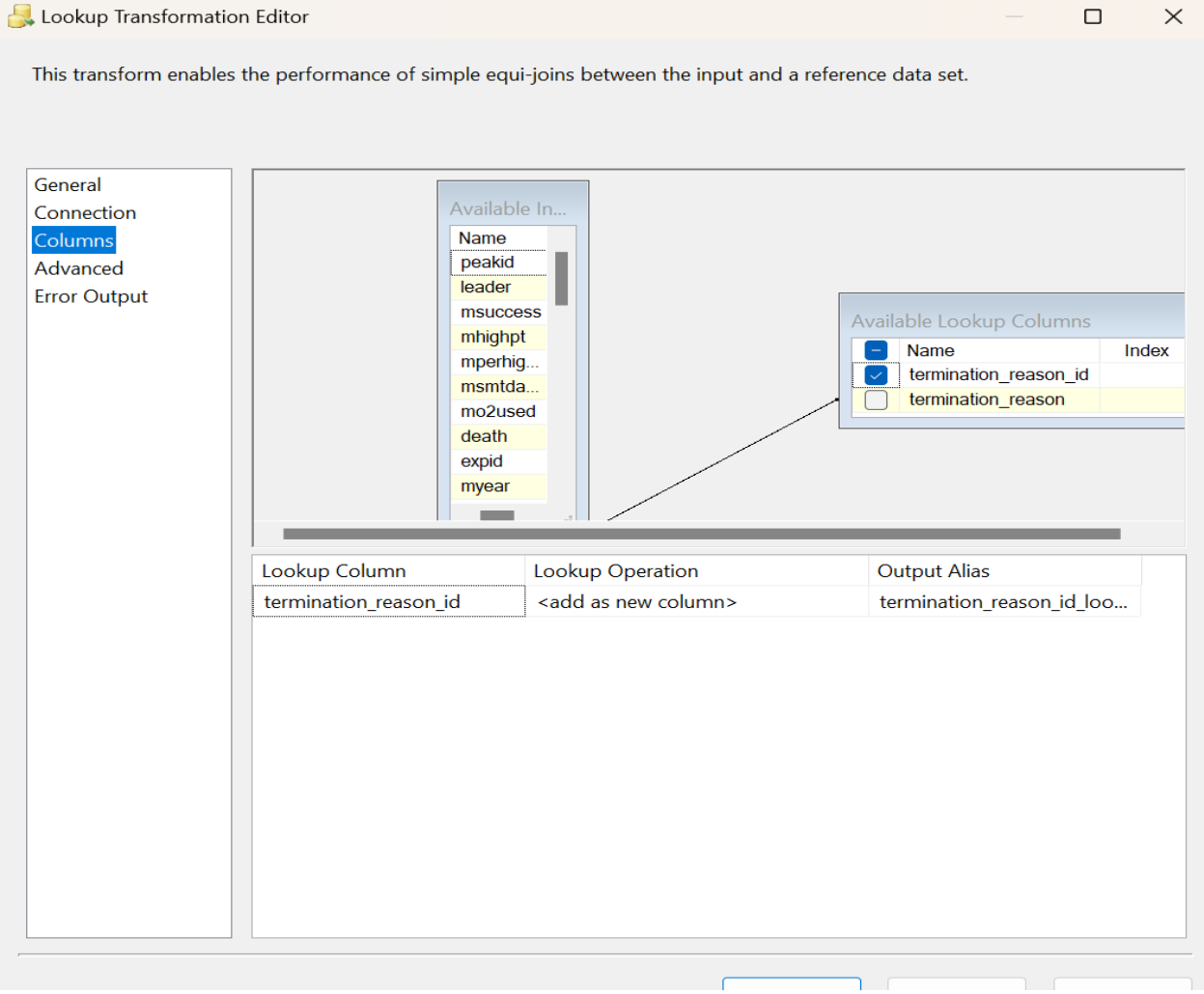
analiziranja podataka i pravljenja izvještaja. Dok *Match Output* šaljemo u komponentu *Union All* koja će nam služiti da spojimo *Match* podatke koje je *Lookup* uspio pronaći i *No Match* podatke koje je *Derived Column* komponenta postavila na –1.



20 – Lookup Nation

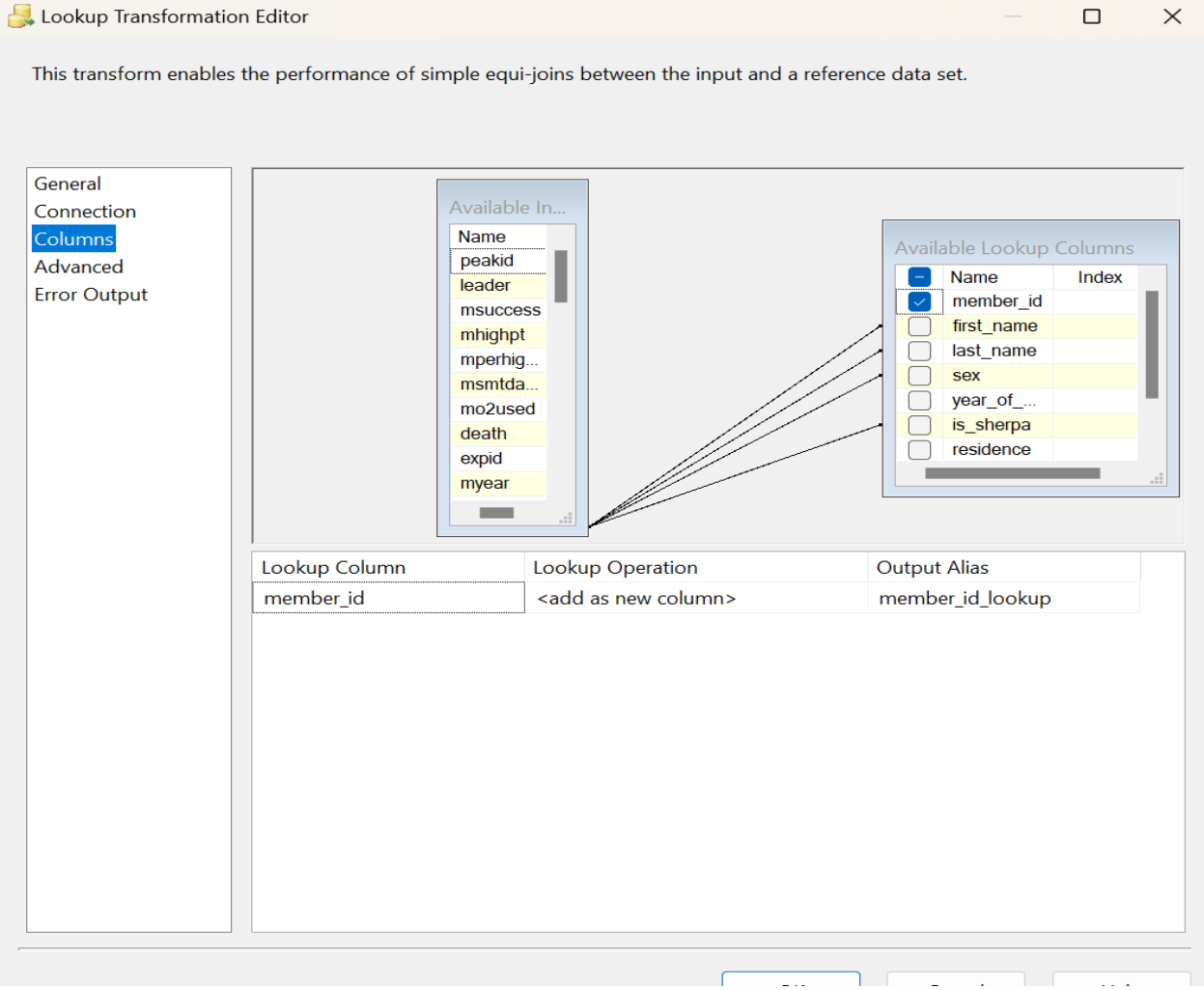
U narednoj *Lookup* komponenti želimo da dođemo do vrijednosti stranog ključa *termination_reason_id* koji će nam služiti da dobijemo podatke o tome zašto je član ekspedicije napustio tu ekspediciju. Ni ovdje nemamo *natural_key*, zbog čega će se pretraživanje željene vrijednosti primarnog ključa vršiti po vrijednosti obilježja *termination_reason*. Prilikom punjenja dimenzione tabele *DimTerminationReason* smo upisivali samo jedinstvene vrijednosti i izbacivali duplikate i zato će *Lookup* moći da pronađe odgovarajuću vrijednost u *DimTerminationReason* tabeli. Kada *Lookup* komponenta naiđe na željenu vrijednost *termination_reason* obilježja onda će vratiti vrijednost primarnog ključa *termination_reason_id* koji će se upisivati kao vrijednost stranog ključa u činjeničnoj tabeli. Na taj način dobijamo i

neophodne podatke o razlogu zašto je učesnik u ekspediciji napustio taj pohod.



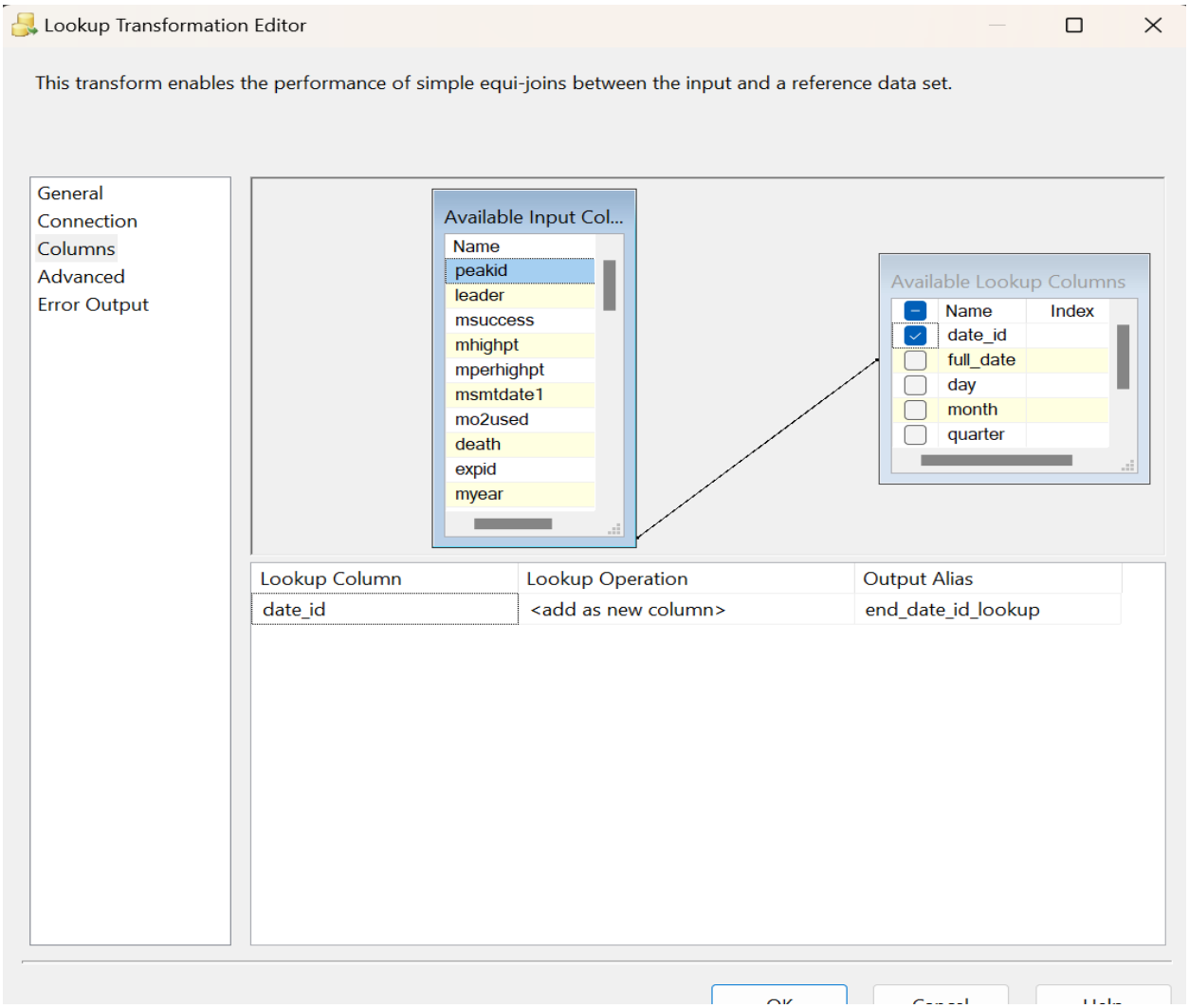
21 – Lookup TerminationReason

Nakon toga vršimo transformaciju određenih podataka u *Derived Column* komponenti koji će nam biti neophodni za ispravan rad naredne *Lookup* komponente. Sada je potrebno naći vrijednost stranog ključa *member_id* koji će nam davati informacije o članu ekspedicije čije se učešće u ekspediciji prati u ovoj činjeničnoj tabeli. Bez *natural_key* opet je neophodno pretraživati članove u *DimMember* dimenzionoj tabeli pomoću vrijednosti drugih obilježja, ali ovaj put će to sada biti kombinacija obilježja, jer ako bi smo koristili samo npr. *fname* za pretragu onda bi smo mogli naći više članova sa tim imenom i *Lookup* komponenta ne zna koji *id* treba da vrati. Iz tog razloga se koristi kombinacija obilježja, isith obilježja na osnovu kojih smo izbacivali grupisane duplikate prilikom punjenja dimenzione tabele *DimMember*, za pretragu vrijednosti primarnog ključa *member_id*. Ta obilježja su *fname, lname, sex i sherpa*. Svaki od članova ima ovu jedinstvenu kombinaciju u *DimMeber* dimenziji i zbog toga *Lookup* komponenta može da vrati jedinstveni *id* za pronađeni red. Na taj način smo dobili sve podatke o članu čije se učešće na ekspediciji prati u datom redu činjenične tabele.



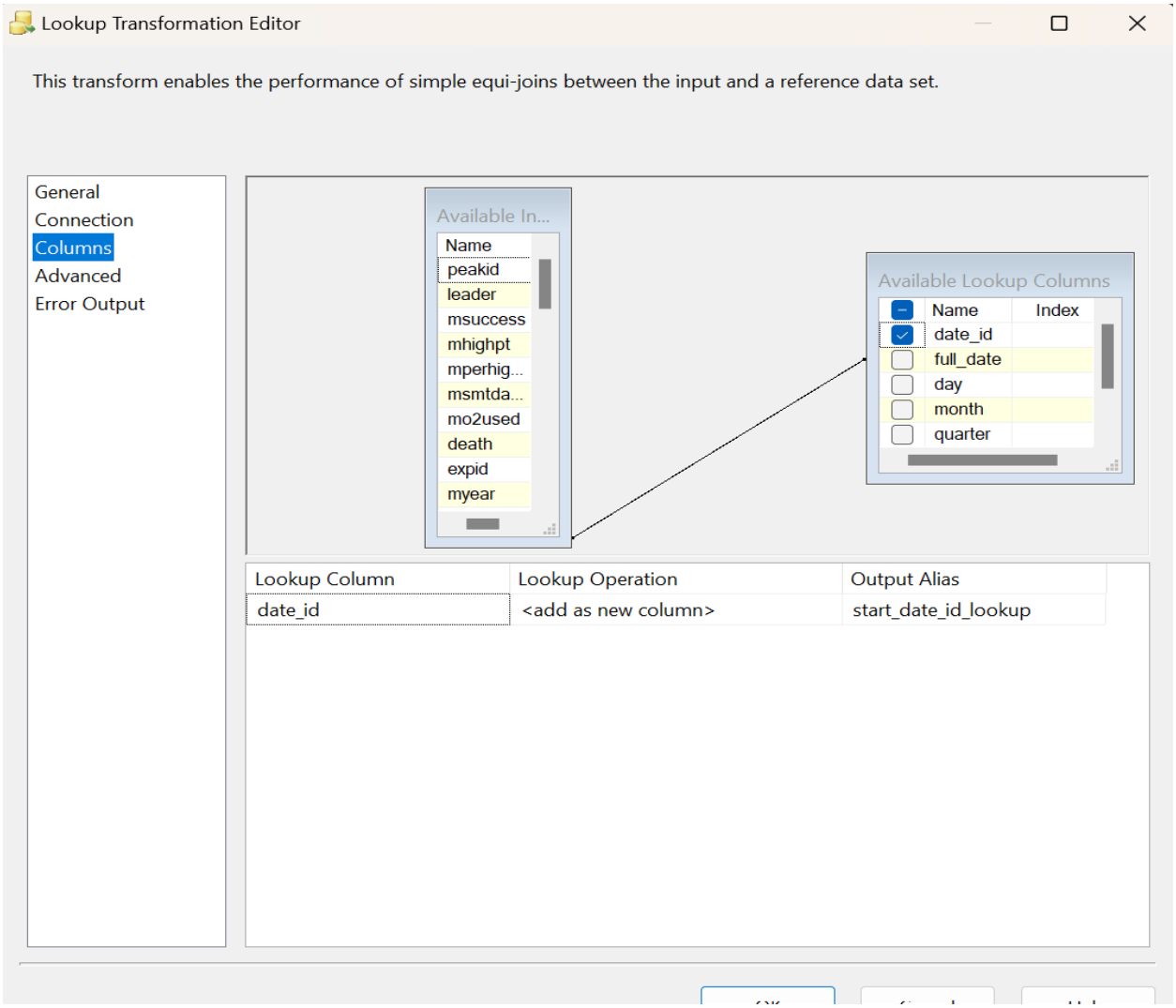
22 – Lookup Member

Svaki od članova ne ekspediciji je na tu ekspediciju iz kampa krenuo iz kampa na isti datum što predstavlja obilježje *bcdate* iz *exped.csv* fajla. Vrijednost tog obilježja je ista za sve članove te ekspedicije. Međutim datum kraja ekspedicije varira od člana do člana, jer je neki učesnik mogao da odustane od ekspedicije prije ili kasnije, to obilježje se nalazi u *members.csv* fajlu i to je *msmdate1*. Takođe može i označavati datum kada je učesnik došao na ciljni vrh ekspedicije ukoliko je uspio u njoj i to je sasvim validan scenario. Prilikom praćenja učešća pojedinačnog člana u nekoj ekspediciji, što je cilj ove činjenične tabele, neophodno je povezati ove podatke sa dimenzionom tabelom *DimDate*. Prilikom pretraživanja spomenute dimenzione tabele u *Lookup* komponenti mi upoređujemo vrijednosti datuma *msmdate1* i *full_date* obilježja iz dimenzione tabele. Svi datumi u vremenskoj dimenziji su jedinstveni i *Lookup* komponenta lako može pronaći odgovarajući datum i da vrati njegov *id*, odnosno virjednost primarnog ključa *date_id*. Ovo će predstavljati vrijednost stranog ključa *end_dat_id* u činjeničnoj tabeli. Na ovaj način smo dobili i datum kada je posmatrani učesnik završio svoju ekspediciju uspješno ili ne. Takođe je opet moguće da vrijednost datum ne postoji ili je prazna pa taj slučaj obrađujemo sa *No Match Output* u *Derived Column* komponentu koja će postaviti vrijednost stranog ključa na –1 što će nam značiti da je vrijednost tog datuma nepoznata. Posle uspješnog obrađivanja *Match* i *No Match Output*-a sada je neophodno da ih povežemo nazad korišćenjem *Union All* komponente.



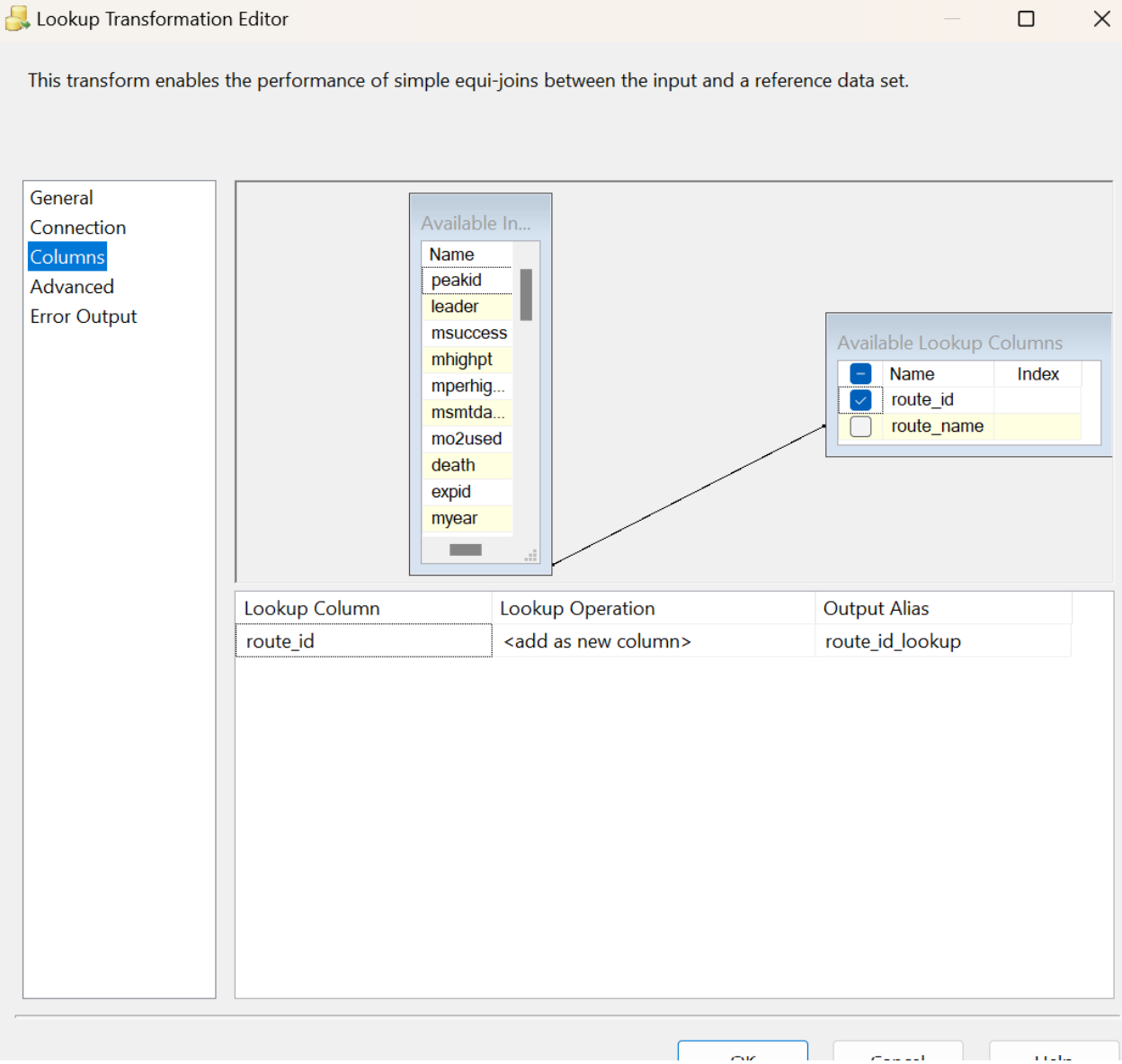
23 – Lookup Date(EndDate)

Analogno i za datum polaska svakog posmatranog člana na posmatranu ekspediciju. Sada ćemo pretraživati samo po drugom obilježju a to je *bcdate* u istoj vremenskoj dimenziji. Na taj način dobijamo informaciju kada su posmatrani članovi pošli iz svojih kampova na ekspediciju. Takođe se nepoznate vrijednosti ili prazne vrijednosti obrađuju na isti način kao i za *end_dat_id*. U ovom slučaju dobijamo vrijednost stranog ključa *start_dat_id* i informaciju kada su učesnici pošli na svoje pohode.



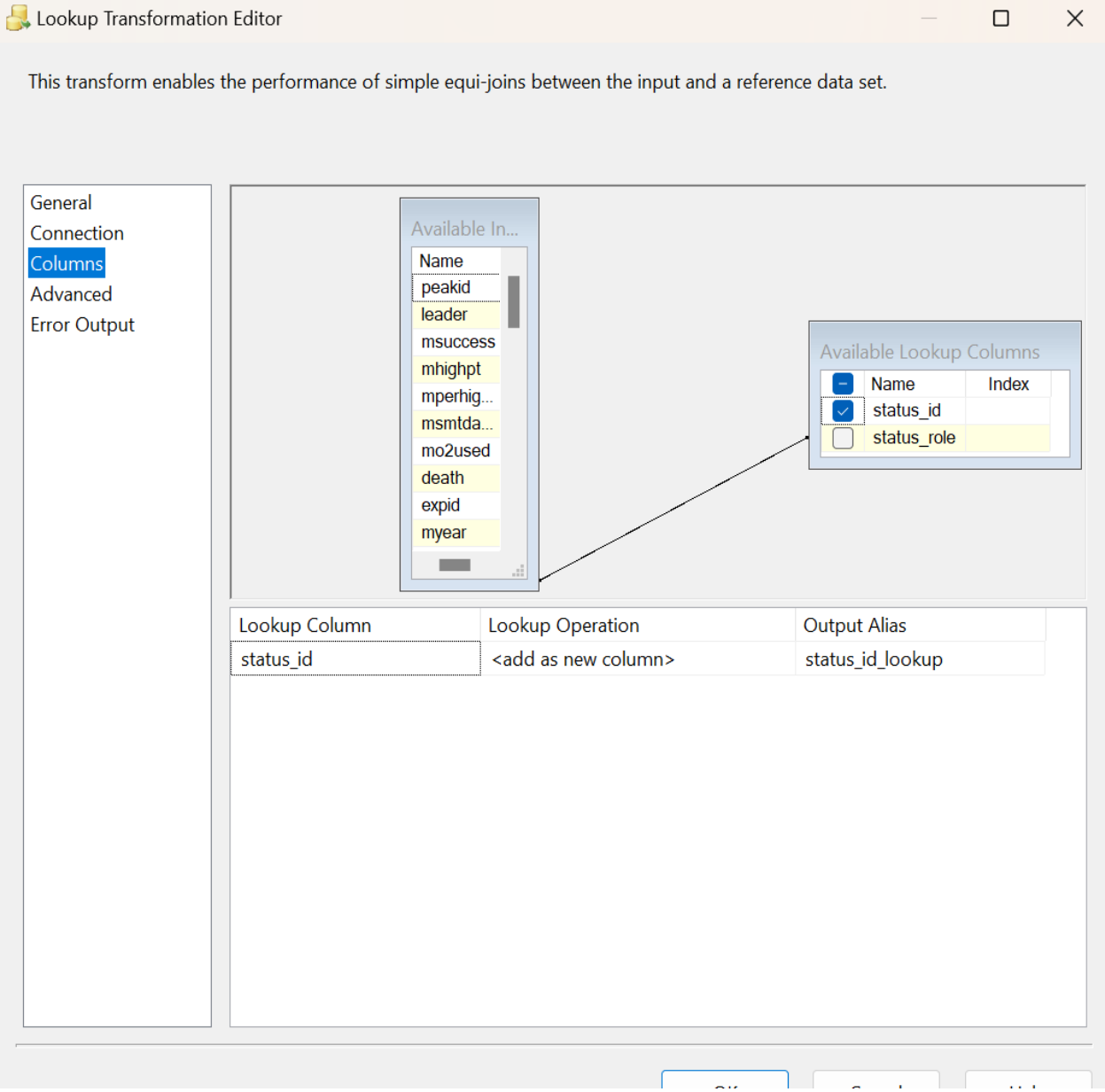
24 – Lookup Date(StartDate)

Svaka od zabilježenih ekspedicija ima i svoju rutu po kojoj su se učesnici kretali. Podaci o tim rutama se nalaze u dimenzionoj tabeli *DimRoute*. Putem *Lookup* komponente pretražujemo datu dimenziju na osnovu *route1* obilježja i tražimo odgovarajuću vrijednost *route_name* obilježja u dimenziji. Nakon toga *Lookup* komponenta vraća *route_id* odnosno virjednost pronađenog primarnog ključa. Takođe opet imamo slučajeve kada nazivi ruta nisu poznati ili su prazni. Takav slučaj je opet riješen putem *No Match Output* u *Derived Column* komponentu koja će vrijednost stranog ključa *route_id* postaviti na –1 što znači da ta vrijednsot ne postoji i da je ne uzimamo u obzir prilikom analize i pravljenja izvještaja. Korišćenjem *Union All* komponente spajamo *Match* i *No Match Output*-e. Na ovaj način dobijamo podatke o ruti po kojoj se kretao posmatrani član ekspedicije u činjeničnoj tabeli.



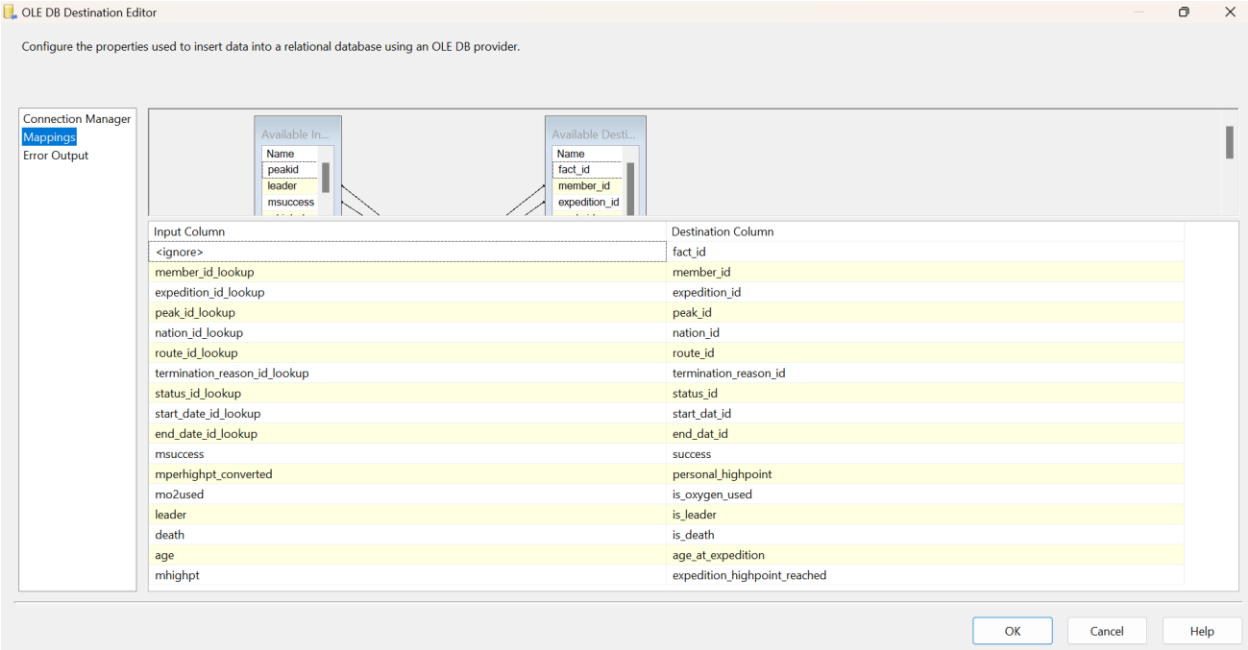
25 – Lookup Route

Svaki od članova ekspedicije ima svoj status ili ulogu u njoj i kada posmatramo učešće pojedinačnog člana u ekspediciji moramo imati i informaciju o tome koja je njegova uloga u toj ekspediciji. Ta informacija se nalazi u dimenzionoj tabeli *DimStatus*. Putem komponente *Lookup* tražimo jedinstveni naziv statusa odnosno *status_role* u dimenziji *DimStatus*, nakon čega nam vraća jedinstvenu vrijednost primarnog ključa tog statusa odnosno *status_id*. Dobijenu vrijednost ćemo postaviti za vrijednost stranog ključa u činjeničnoj tabeli i na taj način dobijamo i neophodnu informaciju o ulozi ili statusu koji je član imao u posmatranoj ekspediciji.



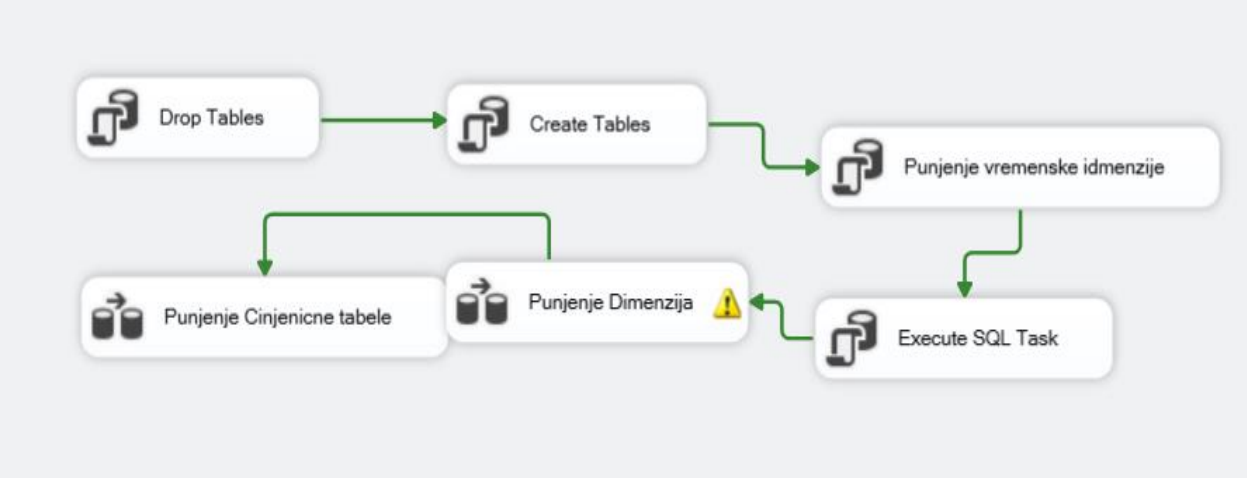
26 – Lookup Status

Konačno dobijanjem vrijednosti svih stranih ključeva i ostalih metrika sada možemo upisati sve podatke u kreiranu činjeničnu tabelu. Kao poslednja komponenta ovog *Data Flow*-a postavljena je *OLE DB Destination* iz razloga jer dobijene podatke moramo upisati u *SQL* bazu podataka, odnosno u činjeničnu tabelu. Sada je neophodno izvršiti određena mapiranja obilježja koja smo doveli u *OLE DB Destination* komponentu i obilježja u bazi podataka. Sva obilježja koja se mapiraju su prethodno već transformisana, pročišćena i konvertovana u određeni tip podatka radi usklađivanja tipova podataka sa definisanim tipovima podataka u bazi.



27 – FactTable Mappings

Na samom kraju slijedi upisivanje svih dobijenih podataka u činjeničnu tabelu čime je cjelokupan *ETL* proces sada završen, svi neophodni podaci su učitani iz izvornih fajlova zatim transformisani i učitani u skladište podataka. Naravno za potrebe konektovanja svih naših podataka sa izvornim fajlovima i sa tabelama u bazi podataka kreirane su posebne konekcije ka njima, koje se koriste od strane svih definisanih komponenti. Ovim je *ETL* proces završen i sada je skladište podataka spremno za sprovođenje analiza i pisanje izvještaja koji će pružiti odgovore na definisana korisnička pitanja i zahtjeve.



28 - Complete DataFlow

6. Prikaz izvještaja

Nakon završenog *ETL* procesa sada na red dolazi analiza dobijenih podataka i kreiranje izvještaja koji će pružiti odgovor na korisnička pitanja i zahtjeve. Za početak prvo je kreiran novi *SSRS* projekat u okviru koga je kreirana konekcija ka skladištu podataka i to će predstavljati osnovni izvor podataka za kreiranje naš izvještaja.

Prvi izvještaj - Uspješnost članova ekspedicija po nacionalnosti

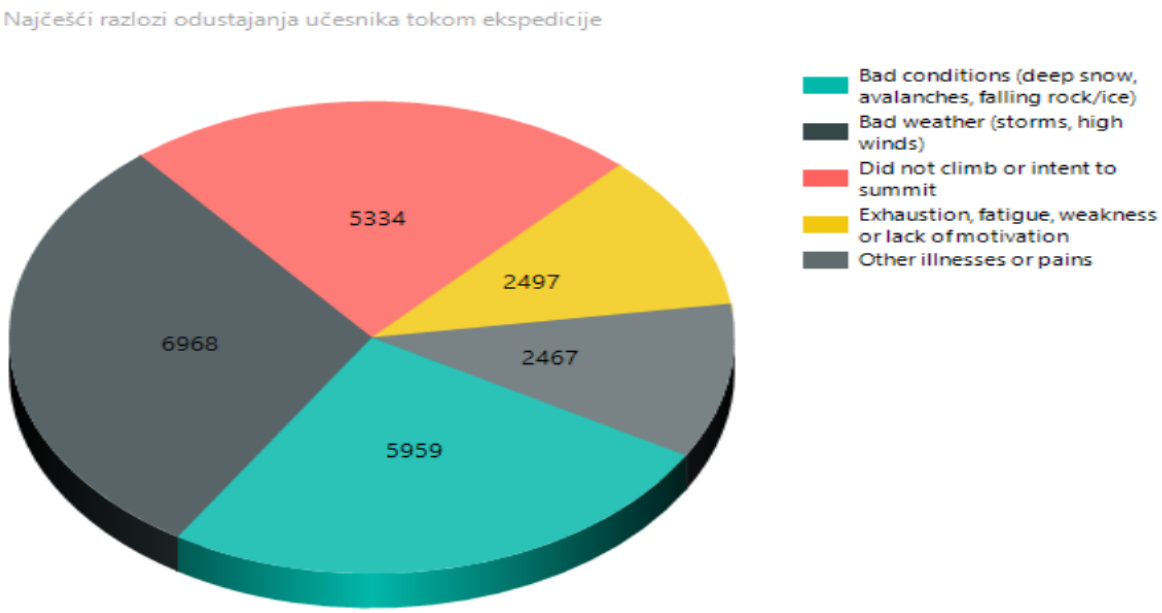
Ovaj izvještaj pruža odgovor na prvo korisničko pitanje i dava uvid našim korisnicima o uspješnosti članova ekspedicija na osnovu njihove nacionalnosti. Time se dobija uvid u to koje su se to nacije najbolje pokazale tokom istorije kao najbolji planinari,

Rang	Nacija	Ukupan Broj Učešća	Broj Uspješnih	Procenat Uspjeha
1	Nepal	20923	14244	68.00%
2	China	2901	1853	64.00%
3	Turkey	59	36	61.00%
4	Iceland	70	39	56.00%
5	Pakistan	78	43	55.00%
6	Kazakhstan	162	86	53.00%
7	Ecuador	159	83	52.00%
8	USSR	271	140	52.00%
9	Ukraine	459	232	51.00%
10	Estonia	56	28	50.00%

29 - Izvještaj 1

Drugi izvještaj - najčešći razlozi za odustajanje od ekspedicije

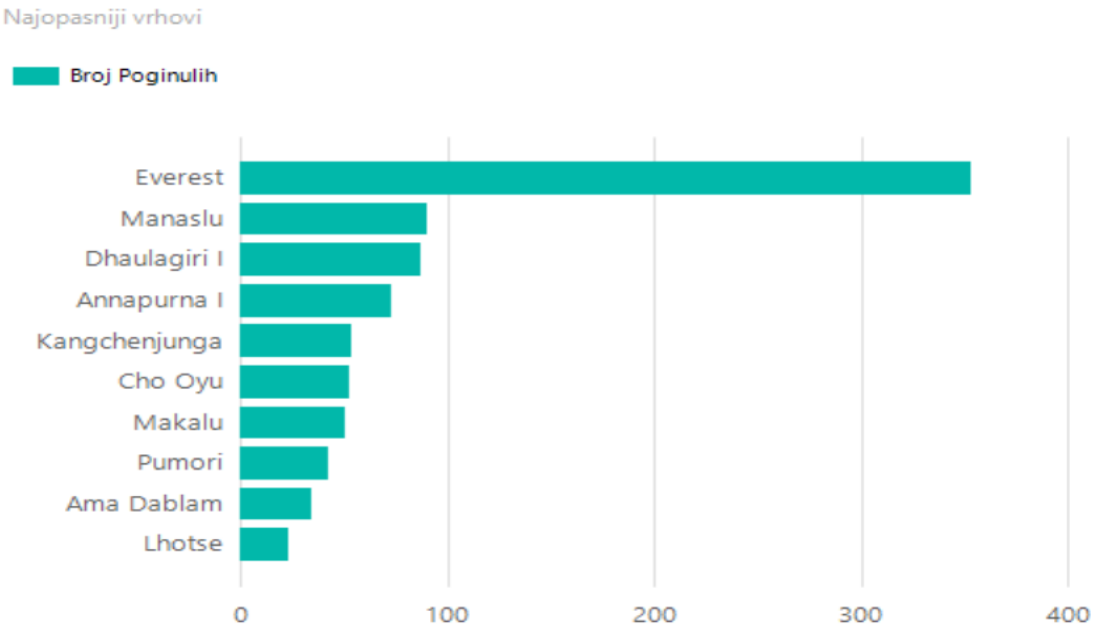
Ovaj izvještaj pruža odgovor na drugo korisničko pitanje i dava im uvid u to koji su to najčešći razlozi zašto su to članovi ekspedicija tokom istorije odustajali od istih i vraćali se u svoje početne kampove. Na taj način može da sazna koji su to razlozi i da se dobro pripremi za njih i u skladu sa tim da se adekvatno pripremi i ponaša prije i tokom same ekspedicije. Izvještaj će prikazivati brojčane vrijednosti koliko se puta je taj razlog javio kroz istoriju kod članova ekspedicija.



30 - Izvještaj 2

Treći izvještaj - Najopasniji vrhovi

Ovaj izvještaj će pružiti odgovor na treće korisničko pitanje i dati našim korisnicima uvid u to koji su to najopasniji vrhovi na *Himalajima*. Kao metriku za tako nešto smo uzeli broj članova ekspedicija koji se nikada nisu vratili sa svojih pohoda na taj ciljni vrh. Na osnovu njega korisnici će znati koji su to najopasniji vrhovi i da se na osnovu toga dobro pripreme za pohod ukoliko se odluče za tako nešto.

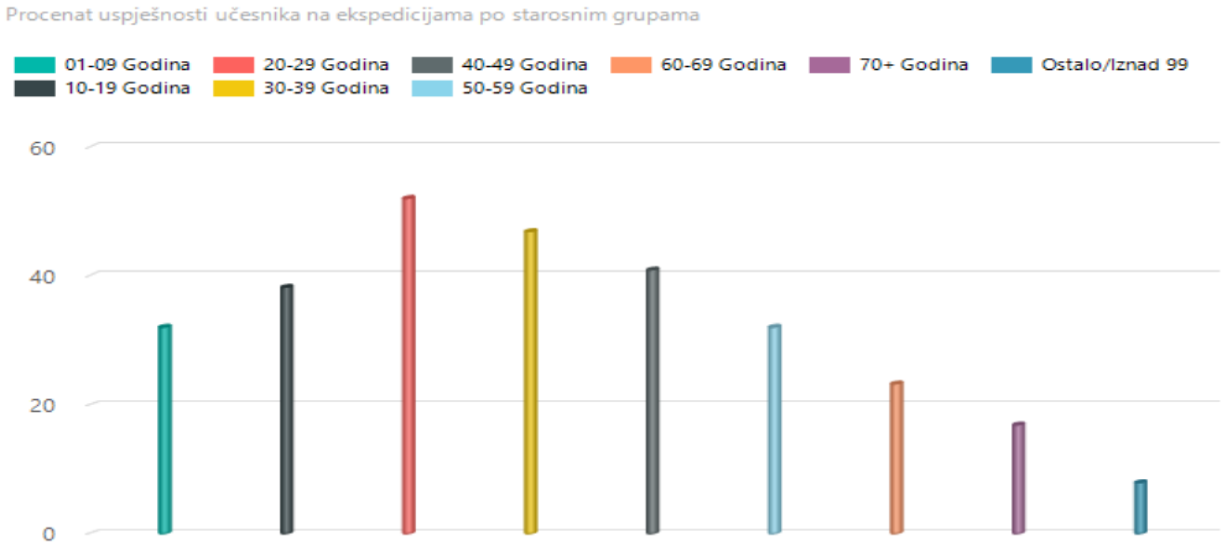


31 - Izvještaj 3

Četvrti izvještaj - Uspješnost članova ekspedicija po starosnim grupama

Ovaj izvještaj pruža odgovor na četvrto korisničko pitanje i pruža našem korisniku uvid u to

kakva je uspješnost članova ekspedicija po starosnim grupama bila kroz istoriju, odnosno u kojim procentima su se članovi ekspedicija uspjevali popeti do ciljnog vrha po starosnim grupama. Ovaj izvještaj omogućava korisniku da identifikuje sebe u starosnim grupama i vidi kojom putanjom se kretala uspješnost kroz starosne grupe i na osnovu toga da obrati više pažnje na kakav se poduhvat sprema i na moguće poslijedice. Takođe može donijeti odluku sa pripadnicima koje starosne kategorije da putuje na osnovu prikazanih rezultata.



32 - Izvještaj 4

Peti izvještaj - Poređenje Šerpasa sa ostalim narodima

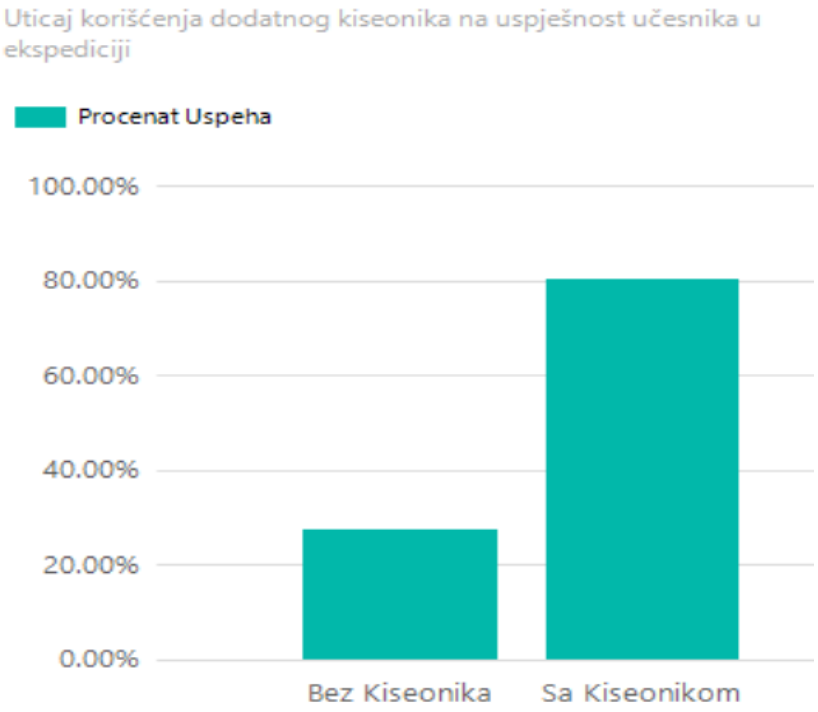
Peti izvještaj daje odgovor na peto korisničko pitanje koji želi da zna koja je to razlika u ovim ekstremnim ekspedicijama između *Šerpasa* i ostalih naroda odnosno nacionalnosti. Korisnik ovim izvještajem dobija uvid u to kako *Šerpasi* prolaze na ekspedicijama, a kako ostali narodi. Na osnovu toga može donijeti odluku da li da angažuje nekog *Šerpasa* na svojoj ekspediciji ili ne, ili čak da se posavjetuje sa njima kako ekspediciji pristupiti maksimalno odgovorno.

Uloga	Ukupno Učešća	Broj Uspješnih	Broj Poginulih	Procenat Uspjeha
Ostali Clanovi	71639	24852	924	34.69%
Šerpasi	17450	12114	247	69.42%

33 - Izvještaj 5

Šesti izvještaj - Uticaj korišćenja dodatnog kiseonika na ishod ekspedicije

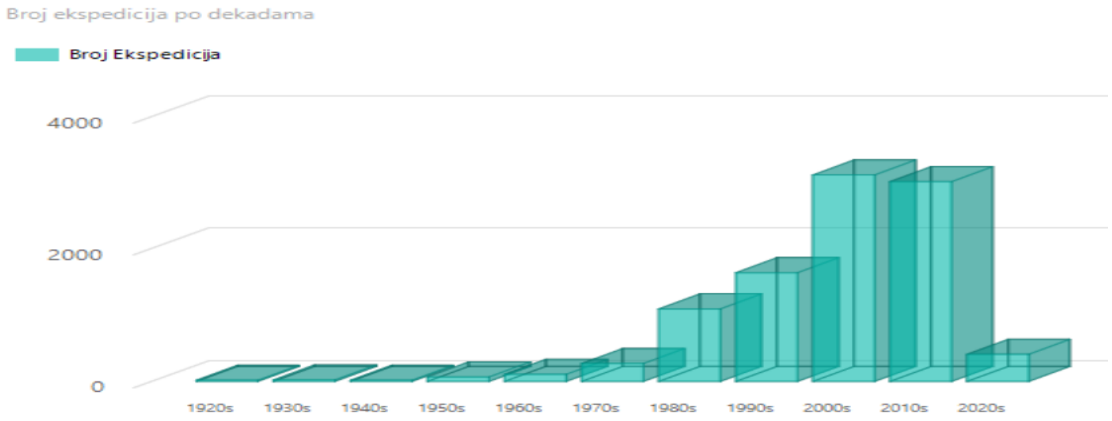
Sledeći izvještaj dava odgovor na šesto korisničko pitanje koje se odnosi na uticaj koji ima korišćenje dodatnog kiseonika na krajnji ishod člana ekspedicije. Ovim izvještajem korsinik dobija uvid u to u kojim procentima su članovi završavali uspješno ekspedicije kada su koristili i kada nisu koristili dodatni kiseonik. Ovim korisnik može da donese odluku o dodatnom snadbjevanju kiseonikom i o njegovom korišćenju i nošenju na ekspedicijama.



34 - Izvještaj 6

Sedmi izvještaj - Popularnost i broj ekspedicija kroz istoriju

Ovaj izvještaj pruža odgovor na sedmo korisničko pitanje kojim korisnik ima uvid u to kada su ekspedicije na *Himalaje* kroz istoriju bile najpopularnije i kada ih je najviše bilo. Podaci su prikazani ukupnim brojem ekspedicija kroz dekade tokom istorije od kada se ovi podaci uopšte i bilježe.



35 - Izvještaj 7

7. Zaključak

Prethodno navedenim koracima kreirano je cjelokupno skladište podataka na osnovu polaznog *dataset*-a. Nakon toga skladište je napunjeno neophodnim podacima na osnovu zadate zvjezdaste šeme. Kako bi smo takođe pružili odgovore na korisnička pitanja i zahtjeve, kreirani su izvještaju unutar *SSRS*-a. Najveći izazov prilikom izrade ovoga projektnog zadatka je bio razumjevanje izvornog *dataset*-a, na koji su način podaci povezani u polaznom *dataset*-u kao i odlučivanje koji nivo granularnosti treba biti postavljen u činjeničnoj tabeli na osnovu prethodno definisanih korisničkih zahtejva. U cilju pružanja odgovora na korisnička pitanja kreirana je zvjezdasta šema sa jednom centralnom činjeničnom tabelom i sa osam dimenzionih tabela. Prilikom rada na ovom projektnom zadatku uočeni su razni nedostatci na polaznom *dataset*-u u smislu nedostajućih ili praznih podataka kao i veliki broj manje važnih metrika i obilježja, tako da bi manje detaljan, u smislu broja obilježja, i više detaljno popunjen polazni *dataset* definitivno pomogao i uticao na performanse kreiranog sistema koje su opet jako dobre. Sam *ETL* proces se izvršava jako brzo što nam omogućava brzo punjenje svih neophodnih tabela nad kojim smo sprovodili neophodne analize i pravili neophodne izvještaje kako bi smo ispunili zadate korisničke zahtjeve. Već kreirani sistem bi se dodatno mogao proširiti i poboljšati postojanjem i korišćenjem kvalitetnije popunjenog *dataset*-a. Neka od dodatnih poboljšanja i povećavanja performansi sitema bi bilo migriranje jednog ovakvog sistema na neko od *cloud* rješenja nad kojim bi se takođe efikasno mogli sprovoditi i učiti *AI* modeli u cilji predikcija ishoda ekspedicija i šta je to potrebno na šta budući član ekspedicije treba da obrati pažnju prilikom odabirta ciljnog vrha. Taj *AI* model bi predlagao koji su to preporučeni vrhovi za starosnu kategoriju učesnika, kako da se pripremi na ekspediciju, na šta treba da obrati pažnju čime da se opremi, kada i kako da se uputi na takav poduhvat. Korišćenjem dobro dizajniranog *cloud* rješenja za jedan ovakav sistem kao i dobro izdefinisan i kreiran *AI* model bi u velikoj mjeri povećao korisničko iskustvo ovakvog sistema kao i performanse ovog sistema.