Productivity Suite

Smart Flow examples

Index

Document control				
Security level	Public			
Company	Amadeus IT Group SA			
Department	Operations			
Author	Dorthe Knight & Nicola Long			
Reviewed by	DKNIGHT		Date	[dd/mm/yyyy]
Approved by	DKNIGHT		Date	[dd/mm/yyyy]
Version	Date	Change	Comment	By
00	09/02/2018	[Text]	[Text]	[Text]

01	08/04/2024	Added example 2 to Credit Card		DKNIGHT

Smart Flow Examples

Add retention line + 185 days (before EoT)

```
send "RTA"
capture line:2, column:32, length:1 assign to var0
capture line:2, column:15, length:5 assign to var1
send "DD" + var1 + "/185"
capture line:2, column:4, length:5 assign to var2
send "RU 1A HK" + var0 + "LON" + var2 + "/KEEP PNR LIVE"
```

Comments: This Smart Flow must be used **before** the PNR has been ended (before EoT). A suggestion would be to use a Smart Trigger to trigger this Smart Flow when adding any PNR element that will only be added once to the PNR at time of creation.

Add retention line + 185 days (after EoT)

```
send "RTA"
capture line:3, column:32, length:1 assign to var0
capture line:3, column:15, length:5 assign to var1
send "DD" + var1 + "/185"
capture line:2, column:4, length:5 assign to var2
send "RU 1A HK" + var0 + "LON" + var2 + "/KEEP PNR LIVE"
```

Comments: This Smart Flow must be used **after** the PNR has been ended (before EoT). A suggestion would be to use a Smart Trigger to trigger this Smart Flow when adding any PNR element that will only be added once to the PNR after the PNR has been created (after EoT).

Add AP element with a call to a personal Smart Flow

```
append "AP 0843 509 0000 " to commandline
call "User name"
append "" to commandline
send commandline
```

Comments: A call when run in the Smart Flow will check both office and personal Smart Flows for a name match.

append "" to commandline is needed if there are no additional appends after the call and before the send commandline.

Called personal Smart flow

```
append "Amadeus" to commandline
```

Master Pricer Expert

```
choose "MASTER PRICER EXPERT
Journey type?" {
  when ("One-way") {
    group{
      mandatory ask "MASTER PRICER EXPERT ONE-WAY
      Departure city e.g. LHR" with format "\b[a-zA-Z]{3}\b" assign to City
      mandatory ask date "Departure date e.g. 10DEC" with format DDMON assign to DepDate
      ask "Departure or arrival time e.g. D1800 or A2100" with format "\b[D|d|A|a]{1}[0-
9]{4}\b" assign to ADTime
      mandatory ask "Destination city e.g. FRA" with format "\b[a-zA-Z]{3}\b" assign to
Destination
    }
    append "FXD" + City + "/D" + DepDate to commandline
    if (ADTime != "") {
      append "/T" + ADTime to commandline
    }
    append Destination to commandline
  }
  when ("Return")    {
    group{
      mandatory ask "MASTER PRICER EXPERT RETURN
      Departure city e.g. LHR" with format "\b[a-zA-Z]{3}\b" assign to CityRT
      mandatory ask date "Departure date e.g. 10DEC" with format DDMON assign to
DepartureRT
      ask "Departure or arrival time e.g. D1800 or A2100 or leave blank" with format
"\b[D|d|A|a]{1}[0-9]{4}\b" assign to ADTimeRT
      mandatory ask "Destination city e.g. FRA" with format "\b[a-zA-Z]{3}\b" assign to
DestinationRT
      mandatory ask date "Return date e.g. 11DEC" with format DDMON assign to ReturnDateRT
      ask "Departure or arrival time e.g. D1800 or A2100 or leave blank" with format
"\b[D|d|A|a]{1}[0-9]{4}\b" assign to ADTimesRT
      mandatory ask "Destination city e.g. LHR" with format "\b[a-zA-Z]{3}\b" assign to
DestinationCityRT
    }
    append "FXD" + CityRT + "/D" + DepartureRT to commandline
    if (ADTimeRT != "") {
      append "/T" + ADTimeRT to commandline
    }
    append DestinationRT + "/D" + ReturnDateRT to commandline
    if (ADTimesRT != "") {
      append "/T" + ADTimesRT to commandline
    }
    append DestinationCityRT to commandline
  }
  when ("Open-jaw") {
    group{
      mandatory ask "MASTER PRICER EXPERT OPEN-JAW
      Departure city .g. LHR" with format "\b[a-zA-Z]{3}\b" assign to CityOJ
      mandatory ask date "Departure date e.g. 15JUN" with format DDMON assign to
DepartureDateOJ
      ask "Departure or arrival time e.g. D0930 or A2100 or leave blank" with format
"\b[D|d|A|a]{1}[0-9]{4}\b" assign to ADTimeOJ
      mandatory ask "Destination city e.g. MIA" with format "\b[a-zA-Z]{3}\b" assign to
DestinationCityOJ
      mandatory ask "Open jaw departure airport e.g. ORL" with format "\b[a-zA-Z]{3}\b"
assign to DepartureAirportOJ
      mandatory ask date "Open jaw departure date e.g. 17JUN" with format DDMON assign to
OJDepartureDateOJ
      ask "Departure or arrival time e.g. D0930 or A2100 or leave blank" with format
"\b[D|d|A|a]{1}[0-9]{4}\b" assign to ADTimesOJ
      mandatory ask "Destination city e.g. LON" with format "\b[a-zA-Z]{3}\b" assign to
DestinationCitysOJ
    }
    append "FXD" + CityOJ + "/D" + DepartureDateOJ to commandline
    if (ADTimeOJ != "") {
      append "/T" + ADTimeOJ to commandline
    }
    append DestinationCityOJ + "--" + DepartureAirportOJ + "/D" + OJDepartureDateOJ to
commandline
    if (ADTimesOJ != "") {
      append "/T" + ADTimesOJ to commandline
    }
    append DestinationCitysOJ to commandline
  }
  when ("Multi-sector"){
```

```
              group {
      mandatory ask "MASTER PRICER EXPERT MULTI-SECTOR
      Departure city e.g. LHR" with format "\b[a-zA-Z]{3}\b" assign to City
      mandatory ask date "Departure date e.g. 10DEC" with format DDMON assign to DepDate
      ask "Departure or arrival time e.g. D1800 or A2100" with format "\b[D|d|A|a]{1}[0-
9]{4}\b" assign to ADTime
      mandatory ask "Destination city e.g. FRA" with format "\b[a-zA-Z]{3}\b" assign to
Destination
              }
    append "FXD" + City + "/D" + DepDate to commandline
    if (ADTime != "") {
      append "/T" + ADTime to commandline
    }
    append Destination to commandline
              choose "MASTER PRICER EXPERT MULTI-SECTOR
  Add more sectors and select Exit when done" until "Exit"{
                when ("Add sector") {
        group{
                                 mandatory ask date "MASTER PRICER EXPERT MULTI-SECTOR
          Onward departure date e.g. 10DEC" with format DDMON assign to DepDate
          ask "Departure or arrival time e.g. D1800 or A2100 or leave blank" with format
"\b[D|d|A|a]{1}[0-9]{4}\b" assign to ADTime
          mandatory ask "Destination city e.g. FRA" with format "\b[a-zA-Z]{3}\b" assign to
Destination
        }
      append "/D" + DepDate to commandline
      if (ADTime != "") {
        append "/T" + ADTime to commandline
      }
      append Destination to commandline
      }
    }
  }
  when ("Price from PNR") {
    append "FXD" to commandline
  }
}
choose "MASTER PRICER EXPERT
Which fare type?"{
  when ("Published only"){
    append "/R,P" to commandline
  }
  when ("Unifares only"){
    append "/R,U" to commandline
  }
  when ("Published & unifares"){
    append "/R,UP" to commandline
  }
  when ("Inclusive tour (ITX) only"){
    append "/RIIT,U" to commandline
  }
}
choose "MASTER PRICER EXPERT
Fare restrictions?"{
  when ("No penalty information"){
    append "/R,*NPE" to commandline
  }
  when ("Refundable fares"){
    append "/R,*RF" to commandline
  }
  when ("No advance purchase information"){
    append "/R,*NAP" to commandline
  }
  when ("No restriction"){
    append "/R,*NR" to commandline
  }
}
ask "MASTER PRICER EXPERT
Add a corporate code? e.g. 123456 or leave blank" assign to CorpCode
if (CorpCode != "") {
  append "/R,U" + CorpCode to commandline
}
```

```
choose "MASTER PRICER EXPERT
Which class?" {
  when ("Economy only"){
    append "/KM" to commandline
  }
  when ("Premium & Economy") {
    append "/KY" to commandline
  }
  when ("Premium economy only") {
    append "/KW" to commandline
  }
  when ("Business only") {
    append "/KC" to commandline
  }
  when ("First only") {
    append "/KF" to commandline
  }
}
choose "MASTER PRICER EXPERT
Majority cabin?" {
  when ("YES I want the majority of flights in the same cabin(s)") {
    append ",*" to commandline
  }
  when ("NO all flights must be in the same cabin(s)") {
  }
}
ask number "MASTER PRICER EXPERT
Add a 'price to beat'e.g. 750

If no price to beat press next or enter to bypass" assign to PriceTB
if (PriceTB!="") {
  append "/PB" + PriceTB to commandline
}
ask "MASTER PRICER EXPERT
Add up to 3 selected airlines e.g. BA or BA,AF etc.

If no preferred airlines press next or enter to bypass" assign to PrefAirline
if (PrefAirline != "") {
  append "/A" + PrefAirline to commandline
}
ask "MASTER PRICER EXPERT
Exclude an airline from the search e.g. BA

WARNING you can only exclude airlines if you have not selected airlines in the last step.

If no airline to exclude press next or enter to bypass" assign to ExcludeAirline
if (ExcludeAirline != "") {
  append "/A-" + ExcludeAirline to commandline
}
send commandline
```

Comments: This key allows you to perform a Master Pricer Expert search as a;

- One-way
- Return
- Open-jaw
- From a booked itinerary while in a PNR
- The search qualifier options in this key are;
- Class
- Majority class (option to apply or not)
- Price to beat' (fares are only returned up to and including this price)
- Include airlines in the search parameters
- Exclude airlines in the search parameters

You could amend this key by adding/removing further Master Pricer Expert search qualifiers to suit the customers search requirements. For more search qualifiers see HE FXD.

Unticketed PNR LPO report

```
choose "<h1>LPO REPORT
</h1>An LPO search allows you to create a list of passenger names and record locators, which
you can display on your terminal, send to a printer, or place on a queue." until "Exit"{
 when ("Create a new LPO report") {
  choose "<h1>LPO REPORT</h1>
  <b>Choose option</b>" {
   when ("ALL live segment PNR's") {
    ask "<h1>LPO REPORT</h1>
    <b>Office ID</b> <i>e.g. LONFB0123

    Leave blank to select the OID you are currently in</i>" assign to AOID
    if (AOID != "") {
     append "LPO/" + AOID + "/ALL" to commandline
    }
    else {
     append "LPO/ALL" to commandline
    }
   }
   when ("Live segment PNR's with a choice of search parameters") {
    ask "<h1>LPO REPORT</h1>
    <b>Office ID</b> <i>e.g. LONFB0123

    Leave blank to select the OID you are in</i>" assign to AOID
    if (AOID != "") {
     append "LPO/" + AOID + "/" to commandline
    }
    else {
     append "LPO/" to commandline
    }
    choose "<h1>LPO REPORT</h1>
    <b>Choose option</b>" {
     when ("Active air segment status codes e.g. HK") {
      mandatory ask "<h1>LPO REPORT</h1>
      <b>Active air segment status codes</b> <i>e.g. HK or WL, KK or for wildcard
search *K etc.

      Enter between 1 and 3 segment status codes</i>" assign to SegStatus
      append "W(" + SegStatus + ")" to commandline
      choose "<h1>LPO REPORT OPTIONS</h1>
      Choose options, when finished select (x) 'RUN REPORT'" until "RUN REPORT" {
       when ("Travel dates/ranges") {
        mandatory ask "<b>Travel date or date range</b> <i>e.g. 10JAN or 15SEP-20SEP

        Enter between 1 and 3 dates/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to TravelDR
        append "D(" + TravelDR + ")" to commandline
       }
       when ("Airline code") {
        mandatory ask "<b>Airline code(s)</b> <i>e.g. BA or BA,VS etc.

        Enter between 1 and 5 airline codes separated with commas</i>" assign to
AirlineCode
        append "A(" + AirlineCode + ")" to commandline
       }
       when ("Board point") {
        choose "<b>Choose board point option</b> <i>
        Board point city & board point within a country cannot be combined</i>" {
         when ("Board point city") {
          mandatory ask "<b>Board point city</b> <i>e.g. ZRH

          Enter between 1 and 3 board point cities</i>" assign to BoardP
          append "B(" + BoardP + ")" to commandline
         }
         when ("Board point within a country") {
          mandatory ask "<b>Board point within country</b> <i>e.g. GB

          Enter between 1 and 3 board point countries</i>" assign to BoardPC
          append "*B(" + BoardPC + ")" to commandline
         }
```

```
                }
              }
              when ("Off point") {
                choose "Off point option <i>off point city & off point within a country
cannot be combined</i>" {
                  when ("Off point city") {
                    mandatory ask "<b>Off point</b> <i>e.g. JFK

                    Enter between 1 and 3 off point</i>" assign to OffP
                    append "O(" + OffP + ")" to commandline
                  }
                  when ("Off point within a country") {
                    mandatory ask "<b>Off point within country</b> <i>e.g. GB

                    Enter between 1 and 3 off point within a country</i>" assign to OffPC
                    append "*O(" + OffPC + ")" to commandline
                  }
                }
              }
              when ("City pair") {
                choose "Choose" {
                  when ("Non-stop/direct") {
                    mandatory ask "<b>City pair</b> <i>e.g. LHRHKG

                    Enter between 1 and 3 city pairs</i>" assign to CityPair
                    append "P(" + CityPair + ")" to commandline
                  }
                  when ("Non-stop/direct and connections") {
                    mandatory ask "<b>City pair</b> <i>e.g. LHRHKG

                    Enter between 1 and 3 city pairs</i>" assign to CityPairX
                    append "X(" + CityPairX + ")" to commandline
                  }
                }
              }
              when ("Class of service") {
                mandatory ask "<b>Class of service</b> <i>e.g. C

                Enter between 1 and 3 classes</i>" assign to ClassOS
                append "C(" + ClassOS + ")" to commandline
              }
              when ("Flight numbers") {
                mandatory ask "<b>Flight numbers</b> <i>e.g. AF123,987

                Enter between 1 and 3 flight numbers</i>" assign to FlightNo
                append "F(" + FlightNo + ")" to commandline
              }
              when ("Creating agent sign") {
                mandatory ask "<b>Creating agent sign</b> <i>e.g. 1234AB

                Enter between 1 and 3 agent signs</i>" assign to CreateAgtSign
                append "S(" + CreateAgtSign + ")" to commandline
              }
              when ("Creation date/ranges") {
                mandatory ask "<b>Creation date or date range</b> <i>e.g. 12JAN or 16OCT-
19OCT

                Enter between 1 and 3 creation date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to CreateDR
                append "K(" + CreateDR + ")" to commandline
              }
              when ("Group PNR's") {
                mandatory ask "<b>Group name</b>" assign to GroupName
                append "G(" + GroupName + ")" to commandline
              }
              when ("ID in mandatory name field") {
                mandatory ask "<b>ID in mandatory name field</b> <i>e.g. INF

                Enter between 1 and 3 ID's</i>" assign to IDMan
                append "I(" + IDMan + ")" to commandline
              }
              when ("Last modified") {
                mandatory ask "<b>Last modified date or date range</b> <i>e.g. 20NOV or
01JAN-15JAN

                Enter between 1 and 3 last modified date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to LastModDR
                append "L(" + LastModDR + ")" to commandline
              }
              when ("OP element containing date") {
                mandatory ask "<b>OP element containing date</b> <i>e.g. 01JAN or 05DEC-
10DEC

                Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to OPElement
                append "Q(" + OPElement + ")" to commandline
              }
              when ("Passenger account number") {
                mandatory ask "<b>Passenger account number</b> <i>e.g. ABC123

                Enter between 1 and 3 account numbers</i>" assign to AccountNo
                append "R(" + AccountNo + ")" to commandline
              }
              when ("Ticketing data") {
                choose "Choose option" {
                  when ("Ticketed PNR's (TKOK element present)") {
                    mandatory ask "<b>Add date or date range</b> <i>e.g. 28JUN or 17AUG-
18AUG

                    Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to TKOK
                    append "T(FA" + TKOK + ")" to commandline
                  }
                  when ("Un-ticketed PNR's (no FA, FHA & SSR TKNA,FHE & SSR TKNE)") {
                    append "V(FA,TA,TE)" to commandline
                  }
                  when ("Un-ticketed PNR's (TL element present)") {
                    mandatory ask "<b>Add date or date range</b> <i>e.g. 02JUL or 22SEP-
23SEP

                    Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to NOTKOK
                    append "Y(TL" + NOTKOK + ")" to commandline
                  }
                }
              }
              when ("Frequent flyer number") {
                mandatory ask "<b>Frequent flyer number</b> e.g. BA-987654321

                Enter between 1 and 3 frequent flyer numbers</i>" assign to FreqFly
                append "Z(" + FreqFly + ")" to commandline
              }
            }
          }
          when ("Non-active air segment status codes e.g. HX"){
            mandatory ask "<h1>LPO REPORT</h1>
            <b>Non-active air segment status codes</b> <i>e.g. GK or HX,UC,UN etc.

            Enter between 1 and 3 segment status codes</i>" assign to NoSegStatus
            append "E(" + NoSegStatus + ")" to commandline
            choose "<h1>LPO REPORT OPTIONS</h1>
            Choose options, when finished select (x) 'RUN REPORT'" until "RUN REPORT" {
              when ("Travel dates/ranges") {
                mandatory ask "<b>Travel date or date range</b> <i>e.g. 10JAN or 15SEP-20SEP

                Enter between 1 and 3 dates/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to TravelDR
                append "D(" + TravelDR + ")" to commandline
              }
              when ("Class of service") {
                mandatory ask "<b>Class of service</b> <i>e.g. C

                Enter between 1 and 3 classes</i>" assign to ClassOS
                append "C(" + ClassOS + ")" to commandline
              }
              when ("Creating agent sign") {
                mandatory ask "<b>Creating agent sign</b> <i>e.g. 1234AB

                Enter between 1 and 3 agent signs</i>" assign to CreateAgtSign
                append "S(" + CreateAgtSign + ")" to commandline
              }
              when ("Creation date/ranges") {
                mandatory ask "<b>Creation date or date range</b> <i>e.g. 12JAN or 16OCT-
19OCT

                Enter between 1 and 3 creation date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to CreateDR
                append "K(" + CreateDR + ")" to commandline
              }
              when ("Group PNR's") {
                mandatory ask "<b>Group name</b>" assign to GroupName
                append "G(" + GroupName + ")" to commandline
              }
              when ("ID in mandatory name field") {
                mandatory ask "<b>ID in mandatory name field</b> <i>e.g. INF

                Enter between 1 and 3 ID's</i>" assign to IDMan
                append "I(" + IDMan + ")" to commandline
              }
              when ("Last modified") {
                mandatory ask "<b>Last modified date or date range</b> <i>e.g. 20NOV or
01JAN-15JAN

                Enter between 1 and 3 last modified date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to LastModDR
                append "L(" + LastModDR + ")" to commandline
              }
              when ("OP element containing date") {
                mandatory ask "<b>OP element containing date</b> <i>e.g. 01JAN or 05DEC-
10DEC

                Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to OPElement
                append "Q(" + OPElement + ")" to commandline
              }
              when ("Passenger account number") {
                mandatory ask "<b>Passenger account number</b> <i>e.g. ABC123

                Enter between 1 and 3 account numbers</i>" assign to AccountNo
                append "R(" + AccountNo + ")" to commandline
              }
              when ("Ticketing data") {
                choose "Choose option" {
                  when ("Ticketed PNR's (TKOK element present)") {
                    mandatory ask "<b>Add date or date range</b> <i>e.g. 28JUN or 17AUG-
18AUG

                    Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to TKOK
                    append "T(FA" + TKOK + ")" to commandline
                  }
                  when ("Un-ticketed PNR's (no FA, FHA & SSR TKNA,FHE & SSR TKNE)") {
                    append "V(FA,TA,TE)" to commandline
                  }
                  when ("Un-ticketed PNR's (TL element present)") {
                    mandatory ask "<b>Add date or date range</b> <i>e.g. 02JUL or 22SEP-
23SEP

                    Enter between 1 and 3 date/date ranges</i>" with format
"^\d{2}\w{3}$|^\d{2}\w{3}-\d{2}\w{3}$" assign to NOTKOK
                    append "Y(TL" + NOTKOK + ")" to commandline
                  }
                }
              }
              when ("Frequent flyer number") {
                mandatory ask "<b>Frequent flyer number</b> e.g. BA-987654321
```

```
        Enter between 1 and 3 frequent flyer numbers</i>" assign to FreqFly
        append "Z(" + FreqFly + ")" to commandline
       }
      }
     }
    }
   }
   when ("Auxiliary segment PNR's with a choice of search parameters"){
    append "LPO/" to commandline
    select "Select auxiliary segment type" from "Air taxi segment,Manual car
segment,Automatic car segment,Cruise segment,Automatic hotel segment,Manual hotel
segment,Insurance segment,Miscellaneous segment,Surface segment,Rail segment,Tour source
segment,Automatic tour segment" assign to SegTypeAux
    append "N(" to commandline
    if (SegTypeAux == "Air taxi segment"){
     append "ATX" to commandline
    }
    if (SegTypeAux == "Manual car segment" ){
     append "CAR" to commandline
    }
    if (SegTypeAux == "Automatic car segment"){
     append "CCR" to commandline
    }
    if (SegTypeAux == "Cruise segment" ){
     append "CRU" to commandline
    }
    if (SegTypeAux == "Automatic hotel segment"){
     append "HHL" to commandline
     mandatory ask "Six character property code" assign to HHLPropcode
     append HHLPropcode to commandline
    }
    if (SegTypeAux == "Manual hotel segment"){
     append "HTL" to commandline
    }
    if (SegTypeAux == "Insurance segment" ){
     append "INS" to commandline
    }
    if (SegTypeAux == "Miscellaneous segment"){
     append "MIS" to commandline
    }
    if (SegTypeAux == "Surface segment"){
     append "SUR" to commandline
    }
    if (SegTypeAux == "Rail segment"){
     append "TRN" to commandline
    }
    if (SegTypeAux == "Tour source segment"){
     append "TTO" to commandline
    }
    if (SegTypeAux == "Automatic tour segment"){
     append "TUR" to commandline
    }
    group{
     mandatory ask date "Start date e.g. 12JAN"  with format DDMON assign to StartDateAux
     mandatory ask date "End date e.g. 15JAN (optional)"  with format DDMON assign to
EndDateAux
     mandatory ask "City or airport code e.g. LHR" assign to CityAux
    }
    append "." + StartDateAux + "-" + EndDateAux + CityAux + ")" to commandline
   }
  }
  choose "<h1>LPO REPORT</h1>
  How would you like to receive your report" {
   when ("Send to your terminal") {
    append "" to commandline
    send commandline
   }
   when ("Queue to a specific queue/category") {
    group{
     mandatory ask "LPO REPORT
     Queue number e.g. 30" assign to Queue
```

```
          ask "Category number e.g. 7" assign to Cat
        }
        append "-Q" + Queue to commandline
        if (Cat != "") {
          append "C" + Cat to commandline
        }
        send commandline
      }
      when ("Send to your local printer") {
        append "-P" to commandline
      }
    }
  }
  when ("Display an existing LPO report") {
    send "LPS/PS"
    mandatory ask "<b>Display report number</b> <i>e.g. 1<i>" assign to ListNumber
    send "LPS/D" + ListNumber
    mandatory ask "<b>Display PNR number</b> <i>e.g. 002<i>" assign to PNRLineNo
    send "LPS" + PNRLineNo
  }
  when ("Delete an LPO report") {
    send "LPS/PS"
    ask "<b>Delete report number</b> <i>e.g. 1 or 1-2 or leave blank to delete all<i>" assign
to DeleteLPO
    if (DeleteLPO != "") {
      send "LPX/" + DeleteLPO
    }
    else {
      send "LPX/ALL"
    }
  }
}
```

Comments: This key performs an LPO search for PNR’s;

- Within the added date range
- Without FA elements
- With a HK or HL segment status.

You can choose from;

- Display on screen
- Send to default printer
- Place on a queue and category (category not mandatory)

Please be aware this LPO report will not find partially ticketed PNR’s. For more search options see HE LPO.

E-mail itinerary or e-ticket receipt

```
send "RTJ"
ask email "Add the e-mail address" assign to var0
choose "What do you want to do?" {
  when ("Send a passenger itinerary") {
    send "RTI"
    mandatory ask "What sectors would you like to include e.g. 3-4 or 3,5 etc.?" assign to
var1
    send "RF EMAIL SMF"
    send "ER"
    send "ER"
    send "IEP-EML-" + var0 + "/S" + var1
  }
  when ("Send passenger e-ticket receipt") {
    send "RTN"
    mandatory ask "What passengers would you like to include e.g. 1-2 or 1,3 etc. ?" assign
to var2
    send "RF EMAIL SMF"
    send "ER"
    send "ER"
    send "ITR-EML-" + var0 + "/P" + var2
  }
}
```

Comments: This key sends a passenger itinerary (sector selection possible) or a passenger ticket receipt (passenger selection possible) to the e-mail address entered.

Display PNR History

```
choose "What would you like to see?" until "Finish, I'm done" {
  when ("Entire PNR history") {
    send "RH"
  }
  when ("Step number e.g. 099") {
    mandatory ask "Add the number" assign to var0
    send "RH" + var0
  }
  when ("Itinerary history") {
    send "RHI"
  }
  when ("Air history") {
    send "RHA"
  }
  when ("Hotel history") {
    send "RHH"
  }
  when ("Car history") {
    send "RHC"
  }
  when ("Fare history") {
    send "RHF"
  }
  when ("Historical fare record") {
    send "RHFR"
  }
  when ("Queue history") {
    send "RHQ"
  }
  when ("Invoice & itinerary history") {
    send "RHB"
  }
  when ("General history i.e. SR/SK/OS") {
    send "RHG"
  }
  when ("All remarks") {
    send "RHR"
  }
  when ("Accounting remarks i.e. RM*/AI/FZ") {
    send "RHY"
  }
}
```

Comments: This key displays all or sections of the PNR history from the options presented. There are more options you could add if required (or less if you remove the options from the key). See HE RH. This key will stay open until you select ‘Finish, I’m done’ or press the Stop button.

Add an auxiliary segment

```
choose "What type of auxiliary segment do you want to add? " until "Finish, I’m done" {
  when ("Air") {
    call "Air auxiliary"
  }
  when ("Hotel") {
    call "Hotel auxiliary"
  }
  when ("Car") {
    call "Car auxiliary"
  }
  when ("Tour") {
    call "Tour auxiliary"
  }
}
```

Comments: This key gives you the choice of adding an air, hotel, car or tour auxiliary to a PNR. Behind each choice is a separate ‘slave’ Smart Flow (see below). The idea to build each option separately is so they can be use independently of the master Smart Flow or linked to other Smart Flows if necessary. This key will redisplay until you select ‘Finish, I’m done’ or press the Stop button.

Hotel auxiliary

```
group {
  mandatory ask "<b>ADD PASSIVE HOTEL</b>

  <b>City code</b> e.g. JFK" assign to CITY
  mandatory ask date "<b>Date of arrival</b> e.g. 10JAN" with format DDMON assign to INDATE
  mandatory ask date "<b>Date of departure</b> e.g. 15JAN" with format DDMON assign to OUT-
DATE
  select "<b>Room type</b>" from "SINGLE,TWIN,DOUBLE,KING,STANDARD,MODERATE,SUPERIOR,EXECU-
TIVE,SUITE" assign to ROOMTYP
  mandatory ask "<b>Rate type</b> e.g. RAC (RAC rate)" assign to RATETYP
  mandatory ask "<b>Rate currency</b> e.g. GBP" assign to CUR
  mandatory ask "<b>Daily rate</b> e.g. 150.00" assign to DLYRATE
  select "Is the rate inclusive of Breakfast
  <font color=red>Change to No if not included</font>" from "Yes,No" assign to Breakfast
  mandatory ask "<b>Hotel name</b> e.g. JFK HILTON" assign to HTLNAME
  mandatory ask "<b>Confirmation number</b> e.g. 123456" assign to CONFNO
  ask "<b>Guarantee method</b> e.g. CCVI4929444433331111EXP1017
  <font color=blue>Optional</font>" assign to GTEETYPE
  ask "<b>Special requests</b> e.g HIGH FLOOR
  <font color=blue>Optional</font>" assign to SPECIAL
}
// BS-91299999 replace with your IATA number
append "HU1AHK1" + CITY + INDATE + "-" +  OUTDATE + "/RT-" + ROOMTYP + "/SR-" + RATETYP
+"/" + CUR + DLYRATE + " " + HTLNAME + "/BS-91299999/CF-" + CONFNO to commandline
if (GTEETYPE!=""){
  append "/G-" + GTEETYPE to commandline
}
if (SPECIAL!=""){
  append "/SI-" + SPECIAL to commandline
}
send commandline
```

Comments: Slave key of ‘Add an auxiliary segment’. This key adds a HU 1A HK1 LON 01SEP-03SEP/TEXT line to the PNR along with RM remark lines with the qualifier letter H and an RIR remark. When this key has been completed the ‘Add an auxiliary segment’ key will re-display (unless this key has been used independently of the parent key).

Create a basic Company Profile

```
send "PM"
group {
  mandatory ask "Add company name (max 60 characters)" assign to var0
  mandatory ask "Add account code (max 6 characters)" assign to var1
}
send "PCN/" + var0
send "AIAN" + var1
group{
  ask "List required purchase order, cost centre, job number etc. If not required leave
blank" assign to var2
  ask "Add priority information" assign to var3
  mandatory ask "Enter company telephone number" assign to var4
  mandatory ask "Enter name of main contact" assign to var5
  mandatory ask "Enter email address for main contact" assign to var6
  mandatory ask "Enter phone number for main contact" assign to var7
  select “form of payment" from "NONREF,Other" assign to var8
}
send "PPR/ACCOUNTING CODES REQ: " + var2
if (var3!="") {
  send "PPR/" + var3
}
send "APB-" + var4 + " COMPANY MAIN CONTACT NUMBER"
send "RC MAIN CONTACT: " + var5
send "RC MAIN CONTACTS EMAIL: " + var6
send "RC MAIN CONTACTS PHONE NUMBER: " + var7
if (var8!=" Other") {
  send "FP" + var8
}
group{
  mandatory ask “Business name (max 30 characters)" assign to var9
  mandatory ask "Business address (number and street only)" assign to var10
  ask "Enter second line of address (optional)" assign to var11
  mandatory ask "Business city e.g. London" assign to var12
  mandatory ask "Business postcode" assign to var13
  mandatory ask "Business country e.g. GB" assign to var14
}
append "AB //CY-" + var9 + "/A1-" + var10 + "/" to commandline
if (var11!="") {
  append "A2-" + var11 + "/" to commandline
}
send commandline + "CI-" + var12 + "/ZP-" + var13 + "/CO-" + var14
}
send "PE"
```

Add form of payment

```
choose "What is the form of payment" {
  when ("Cash") {
    send "FP CASH"
  }
  when ("Check") {
    send "FP CHECK"
  }
  when ("Credit card") {
    call "Credit card"
  }
}
```

Credit card

Example 1:

```
select "Select the card type" from " ,MasterCard,Visa,Amex,Diners,AirPlus" assign to
cardType
if (cardType ==" ")   {
 select "You did not select a card type" from " ,MasterCard,Visa,Amex,Diners,AirPlus"
assign to cardType
}
if (cardType !=" ") {
  if (cardType =="Amex") {
    append "AX" to ccType
    group {
      mandatory ask "Enter card number" with format "(37)[0-9]{12}" assign to cardNumber
      mandatory ask date "Enter expiry date" with format  MMYY assign to expDate
      mandatory ask number "Enter security code" assign to secCode
    }
  }
  if (cardType =="MasterCard") {
    append "CA" to ccType
    group {
      mandatory ask "Enter card number" with format "(5)[0-9]{15}" assign to cardNumber
      mandatory ask date "Enter expiry date" with format  MMYY assign to expDate
      mandatory ask number "Enter security code" assign to secCode
    }
  }
  if (cardType =="Visa") {
    append "VI" to ccType
    group {
      mandatory ask "Enter card number" with format "(4)[0-9]{15}" assign to cardNumber
      mandatory ask date "Enter expiry date" with format  MMYY assign to expDate
      mandatory ask number "Enter security code" assign to secCode
    }
  }
  if (cardType =="Diners") {
    append "DI" to ccType
    group {
      mandatory ask "Enter card number" with format "(3)[0-9]{15}" assign to cardNumber
      mandatory ask date "Enter expiry date" with format  MMYY assign to expDate
      mandatory ask number "Enter security code" assign to secCode
    }
  }
  if (cardType =="AirPlus") {
    append "DI" to ccType
    group {
      mandatory ask "Enter card number" with format "(27|29)[0-9]{14}" assign to cardNumber
      mandatory ask date "Enter expiry date" with format  MMYY assign to expDate
      mandatory ask number "Enter security code" assign to secCode
    }
  }
  send "FPCC" + ccType + cardNumber + "/" + expDate + "*cv" + secCode
}
```

Example 2:

```
select "Select the card type" from " ,MasterCard,Visa,Amex,Diners,AirPlus" assign to
cardType
if (cardType ==" ")   {
 select "You did not select a card type" from " ,MasterCard,Visa,Amex,Diners,AirPlus"
assign to cardType
```

```
choose "select FOP from" {
  when ("Manual Input") {
    group {
    ask "Enter credit card type (VI, AX etc.)" assign to ctype
    ask "Enter credit card number (16 digits)" assign to cno
  ask "Enter the expiry date (mmyy)" assign to cmc
}
send "FP CC" + ctype + cno + "/" + cexpiry + "/N" + cmc
  }
  when ("from profile") {
    group {
    ask "Enter Company or Customer profile Record Locator" assign to cprofile
}
send "PBRC/"+cprofile+"*"
  }
}
```

Change FOP for NEGO fare

```
choose "Choose the correct FP for the Nego fare" {
  when ("Credit Card need to be changed to NONREF") {
    send "RTF"
    ask number "What is the number of the FP line?" assign to FPline
    send "XE" + FPline
    send "FPNONREF"
  }
  when ("Credit card is to be changed to CASH") {
    send "RTF"
    ask number "What is the number of the FP line?" assign to FPline
    send "XE" + FPline
    send "FPCASH"
  }
  when ("This is a route deal so Credit Card is permitted OR airline accepts the merchant")
{
    send "RMCredit Card permitted for NEGO fare"
  }
}
```

*Note this SF can be run from the below Quality Monitor rule

Trigger - Ticketing

Check name - Found TC and FP CC, check FP via Smart Flow

```
if (TourCode.count>0
AND FormOfPayment.where(FormOfPaymentText CONTAINS "CC").count>0
AND GeneralRemark.where(FreeFlow CONTAINS "Credit Card permitted for NEGO fare").count==0)
{
  set status BLOCKING
  choose "Select Validate to run Smart Flow" {
  when ("Run Smart 'Flow Correct FOP for NEGO fare'") {
  call "Correct FOP for NEGO fare"
    }
  }
}
```

APIS – Passport only

```
group {
  mandatory ask  "What is the document issuing country e.g. GBR" with format "\w{3}$"
assign to var2
  mandatory ask number "What is the passport number" assign to var3
  mandatory ask "What is the nationality e.g. GBR" with format "\w{3}$" assign to var4
  mandatory ask date "What is the date of birth" with format DDMONYY assign to var5
  select "Gender" from "Male,Female,Male Infant,Female Infant,Undisclosed" assign to var6
  mandatory ask date "What is the expiry date?" with format DDMONYY assign to var7
  mandatory ask "What is the surname" assign to var8
  mandatory ask "What is the first name" assign to var9
  ask "What is the middle name" assign to var10
  mandatory ask number "Which passenger is this for?" assign to var11
  mandatory ask "Which sectors is this for e.g. 2-4" assign to var12
}
if (var6 == "Male") {
  append "M" to varGen
}
if (var6 == "Female") {
  append "F" to varGen
}
if (var6 == "Male Infant") {
  append "MI" to varGen
}
if (var6 == "Female Infant") {
  append "FI" to varGen
}
if (var6 == "Undisclosed") {
  append "U" to varGen
}
send "SRDOCS YY HK1-P-" + var2 + "-" + var3 + "-" + var4 + "-" + var5 + "-" + varGen + "-"
+ var7 + "-" + var8 + "-" + var9 + "-" + var10 + "-H/P" + var11 + "/S" + var12
```

APIS - All documents

```
send "RTN,I,G"
group {
  mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>Passenger number</b> <i><font color=\"#787875\">e.g. 1</font></i>"
with format "^\d{1,2}$" assign to PaxNum
  ask "<b>Sector number(s)</b> <i><font color=\"#787875\">e.g. 3,5 or leave blank to apply
to all sectors</font></i>" with format "^\d{1,2}$|^\d{1,2}\-\d{1,2}$|^\d{1,2}\,\d{1,2}$"
assign to SectorNos
  mandatory ask "<b>First given name</b> <i><font color=\"#787875\">e.g. DAVID</font></i>"
with format "^[a-zA-Z\s]{2,}\b$" assign to FirstName
  ask "<b>Second name(s)</b> <i><font color=\"#787875\">e.g. RICHARD</font></i>" with
format "^[a-zA-Z\s]{2,}\b$" assign to MiddleName
  mandatory ask "<b>Surname</b> <i><font color=\"#787875\">e.g. SMITH</font></i>" with
format "^[a-zA-Z\s]{2,}\b$" assign to Surname
  mandatory ask date "<b>Date of birth</b> <i><font color=\"#787875\">e.g.
10JAN77</font></i>" with format DDMONYY assign to DateBirth
  select "<b>Gender</b>" from "Male,Female,Male Infant,Female Infant,Undisclosed Gender"
assign to Gender
  select "<b>Document type</b>" from "None (basic info only),Passenger Passport,Approved
Identity Document,Passport Card,Crew Member Certificate,A-Identity Card,C-Identity Card,I-
Identity Card" assign to TravelDocType
}
if (TravelDocType != "None (basic info only)"){
  group{
    mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>Document number</b> <i><font color=\"#787875\">e.g. for passport
958741852</font></i>" with format "^[a-zA-Z0-9]{2,}$" assign to DocNum
    select "<b>Select nationality</b>" from "United Kingdom,Ireland,Other" assign to Natio
    select "<b>Select issuing country</b>" from "United Kingdom,Ireland,Other" assign to
IssuCountry
```

```
    mandatory ask date "<b>Expiry date</b> <i><font color=\"#787875\">e.g.
10JAN19</font></i>" with format DDMONYY assign to ExpDate
  }
}

if (Gender == "Male"){
  append "M" to gen
}
if (Gender == "Female"){
  append "F" to gen
}
if (Gender == "Male Infant"){
  append "MI" to gen
}
if (Gender == "Female Infant"){
  append "FI" to gen
}
if (Gender == "Undisclosed Gender"){
  append "U" to gen
}

if (TravelDocType == "Passenger Passport"){
  append "P" to docType
}
if (TravelDocType == "Approved Identity Document"){
  append "F" to docType
}
if (TravelDocType == "Passport Card"){
  append "IP" to docType
}
if (TravelDocType == "Crew Member Certificate"){
  append "AC" to docType
}
if (TravelDocType == "A-Identity Card"){
  append "A" to docType
}
if (TravelDocType == "C-Identity Card"){
  append "C" to docType
}
if (TravelDocType == "I-Identity Card"){
  append "I" to docType
}

if (Natio == "United Kingdom"){
  append "GBR" to nationality
}
if (Natio == "Ireland"){
  append "IRL" to nationality
}

if (Natio == "Other") {
 mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>As you have chosen 'other' enter country of nationality</b> <i><font
color=\"#787875\">e.g. SPAIN</font></i>" with format "^[a-zA-Z\s]{2,}\b$" assign to
nationality
  send "DC " + nationality
    mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>Enter the nationality code</b> <i><font color=\"#787875\">e.g.
ESP</font></i>" with format "^[a-zA-Z\s]{3}\b$" assign to nationality
}

if (IssuCountry == "United Kingdom"){
  append "GBR" to issCountryCode
}
if (IssuCountry == "Ireland"){
  append "IRL" to issCountryCode
}

if (IssuCountry == "Other") {
 mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>As you have chosen 'other' enter issuing country</b> <i><font
color=\"#787875\">e.g. CANADA</font></i>" with format "^[a-zA-Z\s]{2,}\b$" assign to
issCountryCode
  send "DC " + issCountryCode
    mandatory ask "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>Enter the issuing country code</b> <i><font color=\"#787875\">e.g.
CAN</font></i>" with format "^[a-zA-Z\s]{3}\b$" assign to issCountryCode
}

if (Gender == "Male"){
  append "M" to gens
}
if (Gender == "Female"){
  append "F" to gens
}
if (Gender == "Male Infant"){
  append "MI" to gens
}
if (Gender == "Female Infant"){
  append "FI" to gens
}
if (Gender == "Undisclosed Gender"){
  append "U" to gens
}

if (TravelDocType == "None (basic info only)"){
  append "SRDOCS YY HK1-----" + DateBirth + "-" + gen + "--" + Surname + "-" + FirstName to
commandline
  if (MiddleName != ""){
    append "-" + MiddleName to commandline
  }
  if (MiddleName == ""){
    append "" to commandline
  }
  append "/P" + PaxNum to commandline
  if (SectorNos != ""){
    append "/S" + SectorNos to commandline
  }
  send commandline
}

else {
  append "SRDOCS YY HK1-" + docType + "-" + issCountryCode + "-" + DocNum + "-" +
nationality + "-" + DateBirth + "-" + gen + "-" + ExpDate + "-" + Surname + "-" + FirstName
to commandline
  if (MiddleName != ""){
    append "-" + MiddleName to commandline
  }
  if (MiddleName == ""){
    append "" to commandline
  }
  append "/P" + PaxNum to commandline
  if (SectorNos != ""){
    append "/S" + SectorNos to commandline
  }
  send commandline
}

choose "<h1><font color=\"#005EB8\">Advance Passenger Information System
(APIS)</font></h1><font color=\"#00A9E0\">Add Secure Flight Passenger Data to your
PNR</font><br><br><b>Add SFPD for another passenger?</b>" until "No" {
  when ("Yes") {
    call "APIS variable cleaner"
    call "APIS"
  }
}
```

APIS variable cleaner

```
send "RT"
capture line : 1, column : 40, length : 1 assign to gen
capture line : 1, column : 40, length : 1 assign to docType
capture line : 1, column : 40, length : 1 assign to nationality
capture line : 1, column : 40, length : 1 assign to issCountryCode
capture line : 1, column : 40, length : 1 assign to gens
capture line : 1, column : 40, length : 1 assign to commandline
```

Comments: this clears all data from the append to variables used in the APIS – All documents Smart Flow so that the call “APIS” runs without error on subsequent runs.

Add fare rules

```
send "RTI"
group {
  ask "Which sectors are these fare rules for e.g. 2-3? Leave blank if the rules apply to
ALL sectors" assign to sec0
  select "What currency are the fees" from "GBP,EUR,USD,SEK" assign to cur
  mandatory ask "What is the rebooking charge?" assign to var1
  mandatory ask "What is the minimum stay" assign to var2
  mandatory ask "What is the cancellation charge?" assign to var3
  mandatory ask "What is the maximum stay" assign to var4
}
append "RIR Rebooking charge: " + cur + var1 to commandline
if (sec0!="") {
  append "/S" + sec0 to commandline
}
send commandline
append "RIR Minimum stay: " + var2 to commandline
if (sec0!="") {
  append "/S" + sec0 to commandline
}
send commandline
append "RIR Cancellation charge: " + cur + var3 to commandline
if (sec0!="") {
  append "/S" + sec0 to commandline
}
send commandline
append "RIR Maximum stay: " + var4 to commandline
if (sec0!="") {
  append "/S" + sec0 to commandline
}
send commandline
```

Add fare notes

```
send "TQT"
ask "Add TST fare notes & conditions
If there is more than one TST choose the number, if not press next or enter" assign to var0
if (var0!="") {
  send "TQT/T" + var0
}
capture line : 1, column : 40, length : 12 assign to farg
group {
  mandatory ask "Fare without tax e.g. 350.00" assign to fare
  mandatory ask "Tax e.g. 132.18" assign to tax
  mandatory ask "Change conditions e.g. GBP60.00 to change" assign to chng
  mandatory ask "Refund conditions e.g. no refund allowed " assign to refd
  mandatory ask "Advance purchase conditions e.g. 1 day" assign to apex
  mandatory ask "Minimum stay e.g. saturday night" assign to mins
  mandatory ask "Maximum stay e.g 3 months" assign to maxs
}
send "RIR FARE RULES AND CONDITIONS FOR TST" + var0
send "RIR FARE: GBP" + fare
send "RIR TAX: GBP" + tax
send "RIR CHANGE RULES:"
send "RIR " + chng
send "RIR REFUND RULES:"
send "RIR " + refd
send "RIR ADVANCE PURCHASE:"
send "RIR " + apex
send "RIR MIN STAY: " + mins
send "RIR MAX STAY: " + maxs
send "RIR FARE GUARANTEED UNTIL: " + farg
```

Offers

```
//note [0-9.,-]{1,20} regex will still allow incorrect sector/offer number input 1-2,5,,---
----
choose "<h1><font color=\"#005EB8\">Amadeus Offers</font></h1><font color=\"#00A9E0\">Cre-
ate offers and proposals</font>" until "Exit" {
  when ("Create") {
    choose "Create Offer" until "Return to main menu" {
      when ("Air") {
        choose "Air" {
          when ("FXX result on Screen") {
            ask "Select mark-up or leave blank for no mark-up" with format "^\d{1,4}$" as-
sign to markUp
            append "OFS/A" to commandline
            if (markUp != "") {
              append "/MK-" + markUp to commandline
            }
            send commandline
            capture line : 1, column : 1, length : 7 assign to warningMessage
            if (warningMessage == "WARNING") {
              append "OFS/A" to commandline
              if (markUp != "") {
                append "/MK-" + markUp to commandline
              }
              send commandline
              send "RTOF"
            }
          }
          when ("create FXX (PTC codes have been added to names)") {
            send "RTIA"
            ask "Select the sector numbers e.g. 2 or 2-3 etc." with format "^[0-9]{1,2}([,-
][0-9]{1,2})*$" assign to sectorNumbers
            send "FXX/S" + sectorNumbers
            capture line : 3, column : 1, length : 8 assign to FXXresponse
            if ( FXXresponse != "LAST TKT") {
              group {
                ask number "Enter pricing number" assign to pricingNumber
                ask "Enter passenger number" assign to passengerNumber
              }
              send "FXT" + pricingNumber + "/P" + passengerNumber
            }
            ask "Select mark-up or leave blank for no mark-up" with format "^\d{1,4}$" as-
sign to markUp
            append "OFS/A" to commandline
            if (markUp != "") {
              append "/MK-" + markUp to commandline
            }
            send commandline
            capture line : 1, column : 1, length : 7 assign to warningMessage
            if (warningMessage == "WARNING") {
              append "OFS/A" to commandline
              if (markUp != "") {
                append "/MK-" + markUp to commandline
              }
              send commandline
            }
            select "Any more pricing numbers and passenger numbers to create offers from"
from "no,yes" assign to moreYesNo
            if (moreYesNo == "yes") {
              send "FXX/S" + sectorNumbers
              capture line : 3, column : 1, length : 8 assign to FXXresponse
              if ( FXXresponse != "LAST TKT") {
                group {
                  ask number "Enter pricing number" assign to pricingNumber
                  ask "Enter passenger number" assign to passengerNumber
                }
                send "FXT" + pricingNumber + "/P" + passengerNumber
              }
              ask "Select mark-up or leave blank for no mark-up" with format "^\d{1,4}$"
assign to markUp
              append "OFS/A" to commandline
              if (markUp != "") {
                append "/MK-" + markUp to commandline
              }
              send commandline
              capture line : 1, column : 1, length : 7 assign to warningMessage
              if (warningMessage == "WARNING") {
                append "OFS/A" to commandline
                if (markUp != "") {
                  append "/MK-" + markUp to commandline
                }
                send commandline
              }
              send "RTOF"
            }
          }
        }
      }
      when ("Hotel from rate selection display") {
        group {
          mandatory ask number "Enter hotel rate number to generate the offer" assign to
hotelRate
          ask "Select mark-up or leave blank for no mark-up" with format "^\d{1,4}$" assign
to hotelRateMarkUp
        }
        if (hotelRateMarkUp != "") {
          send "OFS/H" + hotelRate + "/MK-" + hotelRateMarkUp
        }
        else {
          send "OFS/H" + hotelRate
        }
        send "RTOF"
      }
      when ("Hotel from detailed pricing") {
        group {
          mandatory ask number "Enter hotel rate number to generate the offer" assign to
hotelRate
          ask "Select mark-up or leave blank for no mark-up" with format "^\d{1,4}$" assign
to hotelDetailMarkUp
        }
        if (hotelDetailMarkUp != "") {
          send "OFS/H" + hotelRate + "/MK-" + hotelDetailMarkUp
        }
        else {
          send "OFS/H" + hotelRate
        }
        send "RTOF"
      }
      when ("Car from availability") {
        mandatory ask number "Enter list number from the availability display to generate
the offer" assign to carListNumber
              send "OFS/C" + carListNumber
        send "RTOF"
      }
      when ("Car from the rate features display") {
        send "OFS/C1"
        send "RTOF"
      }
      when ("Car from a car segment display (before end transaction)") {
        mandatory ask number "Enter car segment number" assign to carSegmentNumber
        send "OFS/C" + carSegmentNumber
        send "RTOF"
      }
      when ("Offers Proposal") {
        send "RTOF"
        group {
          ask "Customise the name of the proposal for easy identification (Optional)"
          with format "\b.{1,29}\b" assign to proposalTitle
          mandatory ask "Enter offer number(s) e.g. 1 or 1-3 etc."
          with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to proposalOfferNumber
          ask "Mark up amount (Optional)" with format "\b[0-9.,]{0,10}\b"  assign to
markUpAmount
          ask "Mark down amount (Optional)" with format "\b[0-9.,]{0,10}\b" assign to mark-
DownAmount
        }
        append "PRP/O" + proposalOfferNumber + "/" to commandline
        if (proposalTitle != "") {
          append "T-" + proposalTitle to commandline
        }
        if (markUpAmount != "") {
          append "/MK-" + markUpAmount to commandline
        }
        if (markDownAmount != "") {
          append "/MK-" + markDownAmount to commandline
        }
        //PRP/O[n]/T-title/MK-[amount] or PRP/O[n]/T-title/ML-[amount]
        send commandline
        send "RTPR"
      }
    }
    send "RFOFFERSF;ER"
  }
  when ("Display details") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    choose "Display" {
      when ("Offer details") {
        send "RTOF"
      }
      when ("Offer proposal details") {
        send "RTPR"
        mandatory ask "For which Offer(s) would you like to view the details? e.g. 1 or 1-3
etc."
        with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to proposalOfferDetailNumber
        send "RTPR" + proposalOfferDetailNumber
      }
    }
  }
  when ("RII remarks for the offer") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    group {
      ask "Free text remark" assign to RIIremark
      ask "Enter the sector numbers e.g. 2 or 2-3 etc. if the remark is to be linked"
      with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to sectorNumbers
    }
    if (sectorNumbers != "") {
      send "RII" + RIIremark + "/S" + sectorNumbers
    }
    else {
      send "RII" + RIIremark
    }
  }
  when ("Send") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    send "RFOFFERS;ER"
    choose "Send" {
      when ("Offer Notice") {
        send "RTOF"
        group {
          mandatory ask "Which offer(s) would you like to send? e.g. 1 or 1-3 etc."
          with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to sendOfferNo
          ask "Optional linkage of sectors to the offer
          Enter segment numbers e.g. 2 or 2-3 etc" with format "^[0-9]{1,2}([,-][0-
9]{1,2})*$" assign to OFsegments
          ask email "Override the 'From' email address" assign to senderEmail
          ask email "Override the 'Reply to' email address" assign to replyEmail
        }
        if (senderEmail != "") {
          send "RMZ/CONF*FROM:" + senderEmail
        }
        if (replyEmail != "") {
          send "RMZ/CONF*RPLY:" + replyEmail
        }
        send "RTJ"
        choose "Choose e-mail address" {
          when ("In the PNR") {
            if (OFsegments != "") {
             send "OFN/O"  + sendOfferNo + "*S" + OFsegments
            }
            else {
              send "OFN/O"  + sendOfferNo
            }
          }
          when ("Add an e-mail address") {
            mandatory ask email "Add recipient email address" assign to recipientEmail
            if (OFsegments != "") {
              send "OFN-eml-" + recipientEmail + "/O" + sendOfferNo + "*S" + OFsegments
            }
            else {
             send "OFN-eml-" + recipientEmail + "/O" + sendOfferNo
            }
          }
        }
      }
      when ("Offer Proposal") {
        send "RTPR"
        group {
          mandatory ask "Which offer proposals(s) would you like to send? e.g. 1 or 1-3
etc."
          with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to sendOfferNoProposal
          ask "Optional linkage of sectors to the offer
          Enter segment numbers e.g. 2 or 2-3 etc" with format "^[0-9]{1,2}([,-][0-
9]{1,2})*$" assign to OFsegmentsProposal
          ask email "Override the 'From' email address" assign to senderEmailProposal
          ask email "Override the 'Reply to' email address" assign to replyEmailProposal
        }
        if (senderEmailProposal != "") {
          send "RMZ/CONF*FROM:" + senderEmailProposal
        }
        if (replyEmail != "") {
          send "RMZ/CONF*RPLY:" + replyEmailProposal
        }
        send "RTJ"
        choose "Choose e-mail address" {
          when ("In the PNR") {
            if (OFsegments != "") {
              send "OFN/PR"  + sendOfferNoProposal + "*S" + OFsegmentsProposal
            }
            else {
              send "OFN/PR"  + sendOfferNoProposal
            }
          }
          when ("Add an e-mail address") {
            mandatory ask email "Add recipient email address" assign to recipientEmailProposal
            if (OFsegments != "") {
              send "OFN-eml-" + recipientEmailProposal + "/PR" + sendOfferNoProposal + "*S" + OFsegmentsProposal
            }
            else {
              send "OFN-eml-" + recipientEmailProposal + "/PR" + sendOfferNoProposal
            }
          }
        }
      }
    }
  }
  when ("Cancel") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    send "RTOF"
    mandatory ask "Which offer(s) of offer proposal(s) would you like to cancel? e.g. 1 or 1-3 etc."
    with format "^[0-9]{1,2}([,-][0-9]{1,2})*$" assign to cancelOffer
    send "XE" + cancelOffer
  }
  when ("Verify") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    choose "Verify" {
      when ("Offer") {
        send "RTOF"
        mandatory ask number "Which offer would you like to verify?" assign to verifyOfferNo
        send "OFV" + verifyOfferNo
      }
      when ("Offer Proposal") {
        send "RTPR"
        mandatory ask number "Which offer proposal would you like to verify?" assign to offerProposalVerify
        send "RTPR" + offerProposalVerify
        choose "Verify each offer number in the proposal one by one" until "Done" {
          when ("Offer to be verified") {
            ask number "Which offer would you like to verify?" assign to verifyOfferNoProposal
            send "OFV" + verifyOfferNoProposal
          }
        }
      }
    }
  }
  when ("Confirm") {
    ask "Offer reference number,
    leave blank if offer/proposal record is displayed" assign to offerReference
    if (offerReference != "") {
      send "RT" + offerReference
    }
    choose "Confirm" {
      when ("Offer") {
        send "RTOF"
        ask number "Which offer would you like to confirm?" assign to confirmOfferNo
        send "OFK" + confirmOfferNo
      }
      when ("Offer proposal") {
        send "RTPR"
        mandatory ask "Which offer proposal would you like to confirm?" assign to offerProposalConfirm
        send "RTPR" + offerProposalConfirm
        choose "Confirm each offer number in the proposal one by one" until "Done" {
          when ("Offer to be confirmed") {
            ask number "Which offer would you like to confirm?" assign to con-
firmOfferNoProposal
            send "OFK" + confirmOfferNoProposal
          }
        }
      }
    }
  }
}
```

Display ITA status between two airlines

```
group{
  mandatory ask "Enter the validating carrier e.g. BA" assign to var1
  mandatory ask "Enter the agreement airline e.g. AF" assign to var2
}
send "TGAD-" + var1 + "/" + var2
```

Ticketing Element

```
send "TQT"
capture line : 1, column : 7, length : 9 assign to ListYN
if (ListYN =="IVE TST -") {
  choose "No TST in PNR" {
    when ("Add TKTL") {
      ask date "Enter date for TKTL" assign to Date
      send "TKTL" + Date
    }
    when ("End Smart flow and add TST") {
    }
  }
}
else {
  if (ListYN == "P/S  NAME") {
    send "TQT"
    ask number "Multiple TSTs found please enter line number of the first one in list"
assign to TNumber
    send "TQT/T" + TNumber
  }
  capture line : 1, column : 40, length : 5 assign to LDate
  if (LDate == today) {
    choose "Last date for ticketing is today" {
      when ("Have authorisation for ticketing, send to Queue for ticketing") {
        send "TKTL" + LDate
        send "RFSMARTFLOW"
        send "QE11"
      }
      when ("No authorisation, add TKTL") {
        send "TKTL" + LDate
      }
    }
  }
//The else is managing the actions for LDate not being today.
//giving an option to override the date for the TKTL
  else {
    send "RMX/ LAST DATE OF TICKETING FOR THE TST IS " + LDate
    send "RTI,O,RX"
    ask date "Does the ticket timelimit date need to be changed from the displayed:

    RMX LAST DATE OF TICKETING FOR THE TST IS

    <h2>if yes please enter date, otherwise select Next</h2>" with format DDMON assign to
overrideDate
```

```
    if (overrideDate != "") {
      send "TKTL" + overrideDate
      send "RFTKTLSF"
    }
    else {
      send "TKTL" + LDate
      send "RFTKTLSF"
    }
  }
}
```

Comments: If you do not want to offer an override to the captured last date of ticketing (LDate) then the else statement can simply be:

Else {

send "TKTL" + LDate

send "RFTKTLSF"

}

Add back office data

```
send "RTY"
group{
  ask "Add cost centre if CO missing - must be 1 alpha 3 numeric" assign to CostCentre
  ask "Add batch number if YR missing" assign to BatchNumber
  ask number "Add employee number if EM missing" assign to EmployeeNo
  ask "Add business code if AD missing" assign to BusinessUnit
  ask "Add approver if UD2 missing" assign to Approver
  ask "Add booker if PR missing" assign to Booker
}
if (CostCentre!="") {
  send "RM*CO-" + CostCentre
}
if (BatchNumber!="") {
  send "RM*YR-" +   BatchNumber
}
if (EmployeeNumber!="") {
  send "RM*EM-" + EmployeeNo
}
if (BusinessUnit!="") {
  send "RM*AD-" + BusinessUnit
  send "RM*UD1-" + BusinessUnit
}
if (Approver!="") {
  send "RM*UD2-" + Approver
}
if (Booker!="") {
  send "RM*PR-" + Booker
  send "RM*UD3-" + Booker
}
```

Comments: If any of the boxes are left blank the entry will not be transmitted.

EK Chauffeur service request

```
send "RTN,I"
group{
  mandatory ask "EMIRATES CHAUFFEUR SERVICE REQUEST

  How many cars do you need to request e.g. 2" assign to CarNum
  mandatory ask "Airport code e.g. arrival or departure airport code" assign to AirportCode
  mandatory ask date "Date of service e.g. 25JUN" with format DDMON assign to DateOfService
  select "Select arrival or departure" from "Arrival,Departure" assign to ArivalDeparture
  mandatory ask number "Select segment e.g. 2" assign to Segment
  ask "Free text or leave blank" assign to FreeText
  mandatory ask "Passenger association e.g. 1" assign to PaxAssoc
}
if (ArivalDeparture == "Arrival") {
  append "A" to ArrDep
}
if (ArivalDeparture == "Departure") {
  append "D" to ArrDep
}
append "SU EK NN" + CarNum + AirportCode + DateOfService + "CHFD/" + ArrDep + Segment +
"//" to commandline
if (FreeText!="") {
  append FreeText to commandline
}
append "/P" + PaxAssoc to commandline
send commandline
```

EMD

```
send "RT"
choose "EMD financial impact" {
  when ("Retrieve PNR") {
    ask "What is the locator" assign to Loc
    send "RT" + Loc
    capture line : 1, column : 1, length : 22 assign to CaptureLoc
    if (CaptureLoc == "INVALID RECORD LOCATOR"){
      ask "Invalid locator try again, what is the locator" assign to Loc
    }
    if (CaptureLoc != ""){
      send "RT" + Loc
    }
  }
  when ("Create new PNR") {
    group{
      ask "What is the passengers surname" assign to SurnameN
      ask "What is the passengers first name" assign to FirstN
      select "What is the passengers title" from "MR,MRS,MSTR,MISS,MS,DR,LORD,LADY" assign
to TitleN
    }
    send "NM1" + SurnameN + "/" + FirstN + " " + TitleN
    send "JGD/USN"
    capture line : 8, column : 43, length : 50 assign to Sine
    send "AP"
    send "TKOK"
    send "RM EMD CREATED BY " + Sine + " " + today
  }
  when ("Use the PNR in the command page"){
  }
}
mandatory ask "Which airline are you creating this EMD for e.g. BA, FI etc." assign to
AirlineCode
send "EGSD/V" + AirlineCode + "/RFIC-D"
mandatory ask "Add the four letter service code (see command page) e.g. DPST" assign to
ServiceCode
send "RTI,N"
group {
  ask "Add the departure & arrival airport codes e.g. MANJFK" assign to DepArr
  ask date "Add the date of travel e.g. 10AUG" with format DDMON assign to DoT
  ask "Description e.g. DEPOSIT FOR GROUP" assign to EmdName
}
send "IU" + AirlineCode + "HK1 " + ServiceCode + DepArr + "/" + DoT + "-" + EmdName + "/P1"
send "TMC/V" + AirlineCode
group {
  ask "Coupon value e.g. 500.00" assign to CponVal
  ask "Add endorsements e.g. NON REFUNDABLE (or leave this box blank)" assign to EMDEndo
}
send "TMI/ALON/D" + AirlineCode + "/CV-" + CponVal + "/FGBP" + CponVal
send "TMI/FP-NONREFAGT"
if (EMDEndo !="") {
  send "TMI/FE-" + EMDEndo
}
select "Do you want to issue the EMD now?" from "No,Yes" assign to IssueNow
if (IssueNow == "Yes") {
  send "JGD/USN"
  capture line : 8, column : 43, length : 50 assign to Sine1
  send "RF " + Sine1
  send "TTM/RT"
  send "EWD"
}
if (IssueNow == "No") {
  send "RT"
}
```

Comments: This helps the user create and issue an EMD (RFIC code D = financial impact). This RFIC code is used to created EMD’s for such things as name changes, group deposits, refundable balances

Price

```
send "RTN,I"
mandatory ask "Add sector numbers e.g. 2-3 or 2,4 etc." assign to SectorNo
choose "PRICE"{
  when ("Best buy") {
    choose "PRICE" {
      when ("Store a TST") {
        append "FXB" to commandline
      }
      when ("Do not store a TST") {
        append "FXR" to commandline
      }
    }
    choose "PRICE" {
      when ("In the cabin(s) booked") {
        append "/K" to commandline
      }
      when ("Economy") {
        append "/KM" to commandline
      }
      when ("Premium") {
        append "/KW" to commandline
      }
      when ("Economy & Premium") {
        append "/KY" to commandline
      }
      when ("Business") {
        append "/KC" to commandline
      }
      when ("First") {
        append "/KF" to commandline
      }
    }
  }
  when ("Price as booked") {
    choose "PRICE" {
      when ("Store a TST") {
        append "FXP" to commandline
      }
      when ("Do not store a TST") {
        append "FXX" to commandline
      }
    }
  }
}
choose "PRICE" {
  when ("Published") {
    append "" to commandline
  }
  when ("CAT35") {
    append "/R,U" to commandline
  }
  when ("ITX") {
    append "/RIIT,U" to commandline
  }
  when ("BA hand baggage only") {
    append "/R,U*HBOBA" to commandline
  }
  when ("Corporate") {
    mandatory ask "PRICE
    What is the corporate deal code?" assign to CorpCode
    append "/R,U" + CorpCode to commandline
  }
}
append "/S" + SectorNo to commandline
send commandline
ask "PRICE
If more than one fare offered choose the line number or leave blank" assign to FareLineNo
if (FareLineNo!="") {
  send "FXT" + FareLineNo
}
select "PRICE
Happy with the fare displayed/stored if you want to price again select NO" from "YES,NO"
assign to TryAgain
if (TryAgain == "NO"){
  call "PRICE"
}
if (TryAgain == "YES"){
}
```

Comments: This is to price a PNR and store a TST (if required).

Add Bags

```
send "RTN,I"
group {
  mandatory ask number "Which passenger number is this request for e.g. 2" assign to PaxNo
  mandatory ask "What sector number(s) is this request for e.g. 2-3" assign to Sectors
}
choose "What would you like to add?" {
  when ("Bulky baggage") {
    mandatory ask "Specify number, weight and dimensions of bulky baggage" assign to BulkB
    send " SR BULK - " + BulkB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Cabin baggage requiring seat(s)") {
    mandatory ask "Specify number, weight and dimensions of cabin baggage" assign to CbbgB
    send " SR CBBG - " + CbbgB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Extra seat") {
    mandatory ask "Specify requirement for extra seat" assign to BulkB
    send " SR EXST - " + ExstB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Fragile baggage") {
    mandatory ask "Specify number, weight and dimensions of fragile baggage" assign to
FragB
    send " SR FRAG - " + FragB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Sports equipment") {
    mandatory ask "Specify requirement of sports equipment" assign to SpeqB
    send " SR SPEQ - " + SpeqB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Excess baggage") {
    mandatory ask "Specify number, weight and dimensions of excess baggage" assign to XbagB
    send " SR XBAG - " + XbagB + "/P" + PaxNo + "/S" + Sectors
  }
  when ("Weapons, firearms or ammunition") {
    mandatory ask "Specify weapons, firearms or ammunition requirement" assign to WeapB
    send " SR WEAP - " + WeapB + "/P" + PaxNo + "/S" + Sectors
  }
}
```

Comments: This is to request baggage (outside of normal baggage requirements), extra seats etc.