# ASC Services

**Step 1:**

* Insert into : [AscStatementDataGuHourly] / [AscStatementDataCdpGuParty]
* Fetch SO files:
  + MtAvailibilityData
  + MtMustRunGen
  + MtMarginalPrice
  + MtAscRG
  + MtAscIG

**Step 2:**

Looks like a duplicate SP.

* Fetch:
  + MtAvailibilityData
  + MtMustRunGen
  + MtMarginalPrice

**Step 3:**

Calculate Must run EAG/EPG

Update : AscStatementDataGuHourly

**Step 4:**

Calculate Reduced Generation.

Update: AscStatementDataGuHourly

* AscStatementData\_RG\_EAG
* AscStatementData\_AC\_MOD
* AscStatementData\_RG\_LOCC

**Step 5:**

Calculate Increased Generation.

Update: dbo.AscStatementDataGuHourly

* AscStatementData\_IG\_EAG
* AscStatementData\_IG\_EPG
* AscStatementData\_IG\_UPC

**Step 6:**

Calculate Final RG and IG values.

Update table AscStatementDataGuHourly.

Columns:

* AscStatementData\_RG\_AC
* AscStatementData\_IG\_AC
* AscStatementData\_AC\_total

**Step 7:**

Insert into [AscStatementDataGuMonthly].

Calculate Black start and Generator Start values. These are calculated month wise.

Columns:

* AscStatementData\_GBS\_BSC
* AscStatementData\_GS\_NS
* AscStatementData\_GS\_SC
* AscStatementData\_SC\_BSC
* AscStatementData\_MAC (this is the total compensation value for the month)

**Step 8:**

Insert into [AscStatementDataGenMonthly]

Sum AscStatementDataGuMonthly data. Group By [AscStatementData\_Generator\_Id]

**Step 9:**

Update AscStatementDataGenMonthly from AscStatementDataGuMonthly

Columns:

* AscStatementData\_AC\_Total
* AscStatementData\_SC\_BSC
* AscStatementData\_MAC
* AscStatementData\_MRC
* AscStatementData\_TAC

**Step 11:**

Insert into [AscStatementDataZoneMonthly] from AscStatementDataGenMonthly.

Sum and Group by Congested Zone

**Step B:**

Update table BmeStatementDataMpCategoryHourly.

* Calculate BmeStatementData\_ES column.
* Formula changes based on party category code.

Update AscStatementDataZoneMonthly

* Calculate column AscStatementData\_TD. Sum BmeStatementData\_ES from BmeStatementDataMpCategoryHourly table.
* For specific category codes.
* Group by BmeStatementData\_CongestedZoneID.

**Step C:**

Update BmeStatementDataMpCategoryHourly. Calculate Ratio of compensation.

Columns updated:

* BmeStatementData\_MAC = MH.BmeStatementData\_ES/NULLIF(ZM.AscStatementData\_TD,0)\*ZM.AscStatementData\_MAC
* BmeStatementData\_MRC
* BmeStatementData\_IG\_AC
* BmeStatementData\_RG\_AC
* BmeStatementData\_GS\_SC
* BmeStatementData\_GBS\_BSC
* BmeStatementData\_TC

The aforementioned columns are updated via a formula which uses values from AscStatementDataZoneMonthly

Update BmeStatementDataMpCategoryMonthly.

* Sum values in BmeStatementDataMpCategoryHourly.
* Group by BmeStatementData\_PartyCategory\_Code

Again, Update BmeStatementDataMpCategoryMonthly:

* Use AscStatementDataGenMonthly.
* Sum AscStatementData\_TAC column.
* Group by AscStatementData\_PartyCategory\_Code
* Update BmeStatementData\_TAC

**Step D:**

Insert into AscStatementDataMpZoneMonthly:

* Sum columns in BmeStatementDataMpCategoryMonthly
* Group by BmeStatementData\_PartyType\_Code/Congested zone etc.

**Step E:**

Update AscStatementDataZoneMonthly:

* Use AscStatementDataMpZoneMonthly
* Sum (MZM.AscStatementData\_PAYABLE) as TP of AscStatementDataMpZoneMonthly table
* Sum (MZM.AscStatementData\_RECEIVABLE) as TR of AscStatementDataMpZoneMonthly table
* Group by AscStatementData\_CongestedZoneID
* Update AscStatementDataZoneMonthly. Columns:
  + AscStatementData\_TP
  + AscStatementData\_TR

Again, update AscStatementDataZoneMonthly:

* Use table AscStatementDataMpZoneMonthly. SUM(C.AscStatementData\_ES).
* Update AscStatementDataZoneMonthly. Columns:
  + AscStatementData\_ES\_BS = SUM(C.AscStatementData\_ES).
* Group By AscStatementData\_CongestedZoneID

Again, update AscStatementDataZoneMonthly:

* Use table BmeStatementDataCdpContractHourly. Calculate KE\_ES.
* Update AscStatementDataZoneMonthly. columns:
  + AscStatementData\_KE\_ES
  + AscStatementData\_KE\_EB

Update AscStatementDataMPZoneMonthly:

* Update columns:
  + AscStatementData\_TP\_SOLR = (MH.AscStatementData\_ET \* ZM.AscStatementData\_TP)/NULLIF( ISNULL(ZM.AscStatementData\_ES\_BS,0), 0)
  + AscStatementData\_TR\_SOLR = (MH.AscStatementData\_ET \* ZM.AscStatementData\_TR)/NULLIF( ISNULL(ZM.AscStatementData\_ES\_BS,0) , 0)
* Use table AscStatementDataZoneMonthly. Columns:
  + AscStatementData\_ET
  + AscStatementData\_TP
  + AscStatementData\_ES\_BS
  + AscStatementData\_TR

Insert into AscStatementDataMpMonthly.

* Use AscStatementDataMpZoneMonthly. Sum columns:
  + AscStatementData\_MAC
  + AscStatementData\_MRC
  + AscStatementData\_IG\_AC
  + AscStatementData\_RG\_AC
  + AscStatementData\_GS\_SC
  + AscStatementData\_GBS\_BSC
  + AscStatementData\_PAYABLE + AscStatementData\_TP\_SOLR
  + AscStatementData\_RECEIVABLE + AscStatementData\_TR\_SOLR
* Group by AscStatementData\_PartyType\_Code.

**Step 12:**

Update AscStatementDataMpMonthly:

* Use table AscStatementDataMpMonthly\_SettlementProcess to find predecessor data.