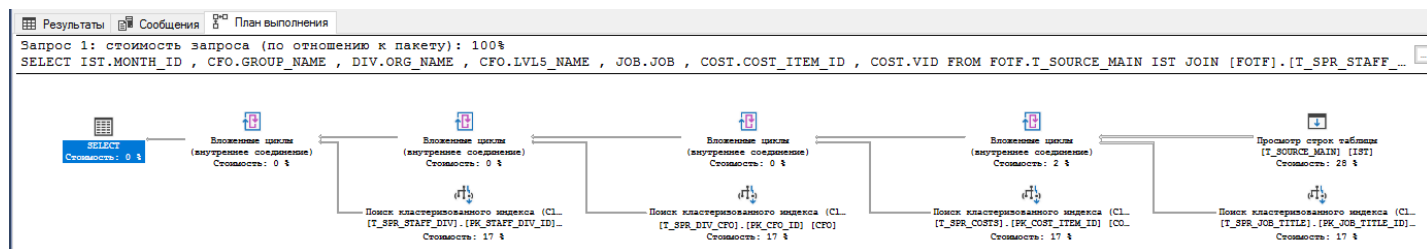


```

SELECT IST.MONTH_ID
      , CFO.GROUP_NAME
      , DIV.ORG_NAME
      , CFO.LVL5_NAME
      , JOB.JOB
      , COST.COST_ITEM_ID
      , COST.VID
FROM FOTF.T_SOURCE_MAIN IST
  JOIN [FOTF].[T_SPR_STAFF_DIV] DIV ON DIV.STAFF_DIV_ID = IST.STAFF_DIV_ID
  JOIN [FOTF].[T_SPR_DIV_CFO] CFO ON CFO.CFO_ID = IST.CFO_ID2
  JOIN [FOTF].[T_SPR_COSTS] COST ON COST.COST_ITEM_ID = IST.COST_ITEM_ID
  JOIN [FOTF].[T_SPR_JOB_TITLE] JOB ON JOB.JOB_TITLE_ID = IST.JOB_TITLE_ID
WHERE MONTH_ID = 202103

```

На данный момент план выполнения такой:



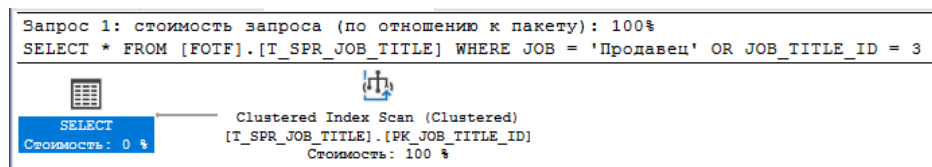
Попробуем пересоздать индексы на составные

Прописываем индексы со вложением для справочников часто используемые поля

```

CREATE NONCLUSTERED INDEX IX_JOB_TITLE_ID_JOB
ON [FOTF].[T_SPR_JOB_TITLE]
(
    [JOB_TITLE_ID] ASC
)
INCLUDE(JOB)
GO

```



Удалим лишние индексы из таблички [FOTF].[T\_SPR\_COSTS] которые создавали ранее

```

DROP INDEX [ind_VID]
ON [FOTF].[T_SPR_COSTS];

DROP INDEX [ind_VID_FUNC_ID]
ON [FOTF].[T_SPR_COSTS];

DROP INDEX [ind_VID_FUNC_ID]
ON [FOTF].[T_SPR_COSTS];

DROP INDEX [ind_VID_UseFOT]
ON [FOTF].[T_SPR_COSTS];

DROP INDEX [ind_VID_UseMSFO]
ON [FOTF].[T_SPR_COSTS];

```

Поменяем на составные индексы

```

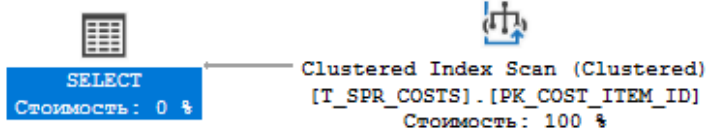
CREATE NONCLUSTERED INDEX IX_SPR_COST_VID_STAT1
ON [FOTF].[T_SPR_COSTS]

```

```
(
    COST_ITEM_ID ASC
)
INCLUDE(Vid,STAT_ID1)
GO
```

Запрос 1: стоимость запроса (по отношению к пакету): 100%

```
SELECT * FROM [FOTF].[T_SPR_COSTS] WHERE VID = 'Оклад по дням' OR COST_ITEM_ID = 8
```



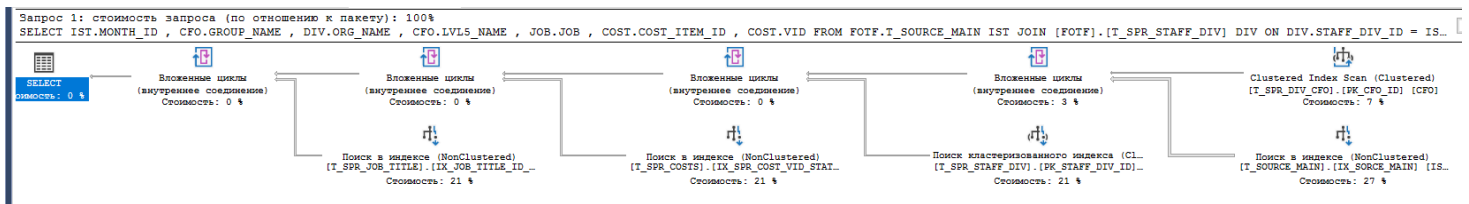
Поменяем на составные индексы

```
CREATE NONCLUSTERED INDEX IX_SORCE_MAIN
ON FOTF.T_SOURCE_MAIN
(
    CFO_ID2
)
INCLUDE(MONTH_ID,STAFF_DIV_ID,COST_ITEM_ID,JOB_TITLE_ID)
GO
```

```
CREATE NONCLUSTERED INDEX IX_SORCE_MAIN_STAFF_DIV_ID
ON FOTF.T_SOURCE_MAIN
(
    STAFF_DIV_ID
)
INCLUDE(MONTH_ID,CFO_ID2,COST_ITEM_ID,JOB_TITLE_ID)
GO
```

```
CREATE NONCLUSTERED INDEX IX_SORCE_MAIN_COST_ITEM_ID
ON FOTF.T_SOURCE_MAIN
(
    COST_ITEM_ID
)
INCLUDE(MONTH_ID,CFO_ID2,STAFF_DIV_ID,JOB_TITLE_ID)
GO
```

Запустим запрос и посмотрим план



Стало только хуже. Видимо в схеме звезда лучше не использовать сложные индексы, хотя может быть нужны более большие таблицы

```
DROP INDEX IX_SORCE_MAIN ON FOTF.T_SOURCE_MAIN;
DROP INDEX IX_SORCE_MAIN_STAFF_DIV_ID ON FOTF.T_SOURCE_MAIN;
DROP INDEX IX_SORCE_MAIN_COST_ITEM_ID ON FOTF.T_SOURCE_MAIN;
DROP INDEX IX_JOB_TITLE_ID_MAIN ON FOTF.T_SOURCE_MAIN;
```

Создадим индексы

```
CREATE INDEX IXT_MAIN_MONTH_ID_ID ON FOTF.T_SOURCE_MAIN (MONTH_ID);
CREATE INDEX IXT_MAIN_CFO_ID2 ON FOTF.T_SOURCE_MAIN (CFO_ID2);
CREATE INDEX IXT_MAIN_STAFF_DIV_ID ON FOTF.T_SOURCE_MAIN (STAFF_DIV_ID);
CREATE INDEX IXT_MAIN_JOB_TITLE_ID ON FOTF.T_SOURCE_MAIN (JOB_TITLE_ID);
CREATE INDEX IXT_MAIN_ATTREBUTE_ID ON FOTF.T_SOURCE_MAIN (ATTREBUTE_ID);
CREATE INDEX IXT_MAIN_COST_ITEM_ID ON FOTF.T_SOURCE_MAIN (COST_ITEM_ID);
```

Запускаем скрипт, теперь как было раньше

Запрос 1: стоимость запроса (по отношению к пакету): 100%

SELECT IST.MONTH\_ID , CFO.GROUP\_NAME , DIV.ORG\_NAME , CFO.LVL5\_NAME , JOB.JOB , COST.COST\_ITEM\_ID , COST.VID FROM FOTF.T\_SOURCE\_MAIN IST JOIN [FOTF].[T\_SFR\_STAFF\_DIV] DIV ON DIV.STA...

