

## **Unit 7. Transient Data & Temporary Storage Control**

---

## Objectives

---

- **Transient Data Control**
- **Types of TDQ's**
- **Commands related to TDQ's**
- **Temporary Storage Control**
- **Commands related to TSQ's**

---

Figure: 7-1. Objectives

**Notes :**

## Transient Data Control

---

- **CICS Transient Data Control Program(TDP) allows a CICS transaction to deal with sequential data called Transient Data Files.**
- **A Transient Data File can be used as either an input or an output file but not both.**
- **The Transient Data is called TDQ( Transient Data Queue ) because the records are put sequentially ( like a queue ).**
- **Control Information of all TDQ's is registered in DCT.**

---

Figure: 7-2. Transient Data Control

### **Notes :**

Transient data queues are identified by a four-character ID called the **destination ID**

Destination Ids and other characteristics of TDQs are defined in the Destination Control Table by the System programmer.

## Types of TDQ's

---

**There are 2 types of TDQ's**

- **Intrapartition TDQ**
- **Extrapartition TDQ**

---

Figure: 7-3. Types of TDQ's

**Notes :**

Two types of Transient Data Queues:

Intrapartition TDQs

Extrapartition TDQs

Static Entries(DCT entry required).

Variable length queue items.

Names 1 – 4 characters.

## Intrapartition TDQ

---

- **Intrapartition TDQ is a group of sequential records which are produced and processed within CICS region.**
- **All Intrapartition TDQ's are stored in only one physical file in a CICS region.**
- **An application program can access the record sequentially and the record will be logically removed from the queue.**
- **Read of a Intrapartition TDQ is destructive.**

---

Figure: 7-4. Intrapartition TDQ

### **Notes :**

All Queues share a VSAM ESDS file.

Destructive read.

Recoverable.

Automatic task Initiation(ATI) capable.

No random access and no updates.

## Extrapartition TDQ

---

**Extrapartition TDQ is a group is a group of records written sequentially which interfaces between transactions of the CICS region and the systems outside the CICS region.**

**Each Extrapartition TDQ is a separate physical file, which means that each file must be open within the CICS region when it is used by the CICS transaction.**

---

Figure: 7-5. Extrapartition TDQ

### **Notes :**

Extra partition TDQs are :

- MVS SAM files.
- Open input or output(not both)
- Sequential access only
- Records may be fixed or variable length , blocked or unblocked
- Records accessible from a batch program

## WRITEQ TD Command

---

```
EXEC CICS WRITEQ TD  
    QUEUE ( name )  
    FROM ( data_area )  
    [LENGTH ( data_value )]  
    [SYSID ( systemname )]  
END-EXEC.
```

---

Figure: 7-6. WRITEQ TD Command

### Notes :

This command writes data to a transient data queue(TDQ)

The transient data queue name is a predefined symbolic destination defined in the DCT(Destination Control Table).

## READQ TD Command

---

```
EXEC CICS READQ TD
      QUEUE ( name )
      INTO ( data_area ) SET ( ptr_ref )]
      [LENGTH ( data_value )]
      [SYSID ( systemname )]
      [NOSUSPEND]
END-EXEC.
```

---

Figure: 7-7. READQ TD Command

### Notes :

This command reads from- a transient data queue(TDQ) after which the record is no longer available.



## DELETEQ TD Command

---

```
EXEC CICS DELETEQ TD
      QUEUE ( name )
      [SYSID ( name )]
      [NOSUSPEND]
END-EXEC.
```

---

Figure: 7-8. DELETEQ TD Command

### Notes :

This command deletes all data from an Intrapartition destination (queue), and releases (deallocates) all storage associated with the destination.

Once a transient data is processed it should be deleted because the space used is still allocated and therefore unavailable for use by other CICS users or transactions.

This command can be used on Extrapartition transient data queue.

## EXCEPTION Conditions

---

- **Special Handling Required**

**QZERO                      LENGERR**

- **Special Handling Desirable**

**NOSPACE                  NOTOPEN                  QBUSY**

- **Error**

**IOERR                      QIDERR**

---

Figure: 7-9. EXCEPTION Conditions.

### Notes :

QZERO Destination empty or end of the transient data queue.

NOSPACE no space on the intrapartition data queue.

NOTOPEN extra partition destination closed.

QBUSY attempt to access a record in an intra partition TDQ that is being written to or deleted by another task.

## Transient Data Input Example

---

### WORKING-STORAGE SECTION.

#### 01 WRKFLDS.

05 TD-RECL PIC S9(4) COMP.

05 TD-ERRCODE PIC S9(8) COMP.

#### 01 TDREC.

05 TD-DATE PIC X(8).

05 TD-NUM PIC X(6).

05 TD-AMOUNT PIC X(8).

.....

.....

### PROCEDURE DIVISION.

MOVE 22 TO TD-RECL.

PERFORM UNTIL TD-ERRCODE=DFHRESP(QZERO)

EXEC CICS READQ TD QUEUE('LSTC')

INTO (TD-REC)

LENGTH(TD-RECL)

RESP(TD-ERRCODE)

END-EXEC.

IF TD-ERRCODE=DFHRESP(NORMAL)

.....

ELSE

IF TD-ERRCODE NOT = DFHRESP(QZERO)

.....

.....

.....

.....

---

Figure: 7-10. Transient Data Input Example.

## Temporary Storage Control

---

- **CICS Temporary Storage Program(TSP) provides the application program with an ability to store and retrieve data in a Temporary Storage Queue(TSQ)**
- **TSQ is a queue of stored records which is created and deleted dynamically by the application programs.**
- **TSQ is identified by queue id and a record in it is identified by a relative position number called item number.**
- **Read on a TSQ is not destructive.**
- **TSQ can be written in the main storage or auxiliary storage.**
- **TSQ can be accessed by any transaction in the same CICS region.**

---

Figure: 7-11. Temporary Storage Control

### **Notes :**

Optionally if you need a recoverable queue, the system programmer must define it in the TST(Temporary Storage Table)

Data in temporary storage is remains available until explicitly purged.

## WRITEQ TS Command

---

```
EXEC CICS WRITEQ TS
  QUEUE ( name )
  FROM ( data_area )
  [LENGTH ( data_area )]
  [NUMITEMS ( data_area ) ITEM ( data_area ) [REWRITE]]
  [SYSID ( systemname ) ]
  [MAIN | AUXILLARY]
  [NOSUSPEND]
END-EXEC.
```

---

Figure: 7-12. WRITEQ TS Command

### Notes :

This command writes temporary data (records) in a temporary storage queue in either main or auxillary storage.

If the write is to a recoverable queue, then after issuing a DELETEQ TS ,no WRITEQ TS can be issued until a sync point has occurred.

## READQ TS Command

---

```
EXEC CICS READQ TS
  QUEUE ( name )
  [INTO ( data_area ) | SET ( ptr_ref )]
  [LENGTH ( data_area )]
  [NUMITEMS ( data_area )]
  [ITEM ( data_area ) | NEXT]
  [SYSID ( systemname )]
END-EXEC.
```

---

Figure: 7-13. READQ TS Command

### Notes :

This command reads data from specified temporary storage queue in main or auxillary storage.

## DELETEQ TS Command

---

```
EXEC CICS DELETEQ TS
      QUEUE ( name )
      [SYSID ( systemname )]
END-EXEC.
```

---

Figure: 7-14. DELETEQ TS Command

### Notes :

This command will delete a temporary storage (TS) queue.

It will delete any data remaining in the TS queue and return all storage used by the TS queue to CICS, which then can be used by another transaction.

## EXCEPTION Conditions

---

- **Special Handling Required**

**ITEMERR**

**LENGERR**

- **Special Handling Desirable**

**NOSPACE**

- **Error**

**IOERR**

**QIDERR**

---

Figure: 7-15. EXCEPTION Conditions.

Notes :

**ITEMERR** the item number specified or implied by a READQ TS command or a WRITEQ TS with **REWRITE** is outside of the range of entry numbers assigned for the queue.

**NOSPACE** there is no space on the auxiliary dataset.



## Temporary Storage Sequential Output Example

---

### WORKING-STORAGE SECTION.

#### 01 WRKFLDS.

05 TS-RECL PIC S9(4) COMP.  
05 TS-ERRCODE PIC S9(8) COMP.  
05 TS-ITEMNO PIC S9(4) COMP.  
05 TS-ID PIC S9(8).

.....

#### 01 TDREC PIC X(80).

.....

### PROCEDURE DIVISION.

STRING EIBTRMID EIBTRNID DELIMITED SIZE INTO TS-ID.

EXEC CICS WRITEQ TS QUEUE(TS-ID)  
FROM (TS-REC)  
LENGTH(TS-RECL)  
ITEM(TS-ITEMNO)  
RESP(TS-ERRCODE)  
END-EXEC.

IF TS-ERRCODE=DFHRESP(NORMAL)  
.....  
ELSE  
IF TS-ERRCODE NOT = DFHRESP(QZERO)  
.....  
.....  
.....  
.....

---

Figure: 7-16. Temporary Storage Sequential Output.