



Getting to know your IMS Scheduler (How do those pesky IMS transactions get scheduled?)

Deepak Kohli
IMS Offering Management
deepakk@us.ibm.com



Central Canada DB2 Users Group (CCDUG), June 5, 2017

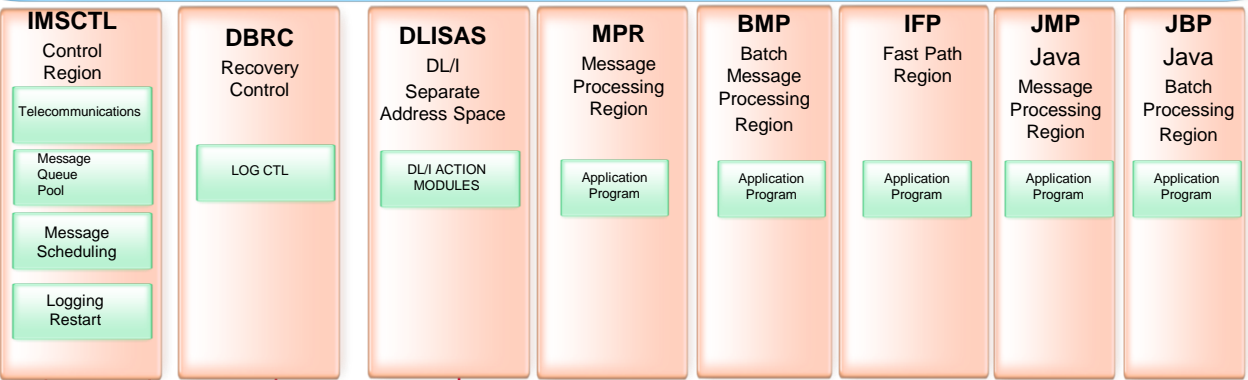
Agenda

- IMS Architecture
- IMS Queuing
 - Class, Priority
 - Limit Priority, Limit Count
 - PROCLIM, Quick Reschedule
- Scheduling
 - MPR classes
- Other scheduling related parameters
 - WFI
 - PWFI
 - Serial vs Parallel scheduling, PARLIM

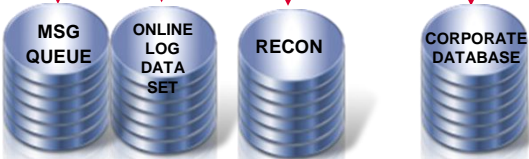
IMS Architecture Overview



z/OS



← Dependent Regions →



→ IMS TM makes data available to the end user

IMS Architecture Overview – TCP/IP input



z/OS

TCP / IP



Scheduling Overview

- Picking the Transaction to process
 - Scheduling path length (can involve serialization)
 - Make sure the resources it needs are available

- Once the transaction is picked
 - Load the application program into the dependent region
 - And give control to the program

Defining Application resources

- IMSGEN or DRD (Dynamic Resource Definition)
- IMSGEN Macros:
DATABASE DBD1
DATABASE DBD9
APPLCTN PSB=PROGRAM1, ...
 TRANSACTION CODE=TRANX, ...
 TRANSACTION CODE=TRANY, ...
APPLCTN PSB=PROGRAM2, ...
 TRANSACTION CODE=TRANW, ...
- APPLCTN & TRANSACTION macros contain scheduling parms

Transaction Class & Priority

Transaction Classes & Priorities

- Every transaction belongs to a class & has a priority within that class
- Class can be specified on the APPLCTN or TRANSACT macro:

```
APPLCTN   PSB=PROG1, ..., PGMTYPE=(TP,,class)  
          TRANSACT CODE=XYZ, ..., MSGTYPE=(,,class)
```
- Class can be dynamically changed via /ASSIGN command

Transaction Class & Priority

CLASSES

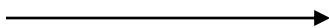
TRANSACTIONS (PRIORITY)

004



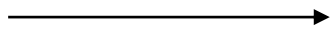
SKILLUPD (8)
SKILLINQ (5)
PAYROLL (4)

002



INVENT (10)
STOKSTAT (8)

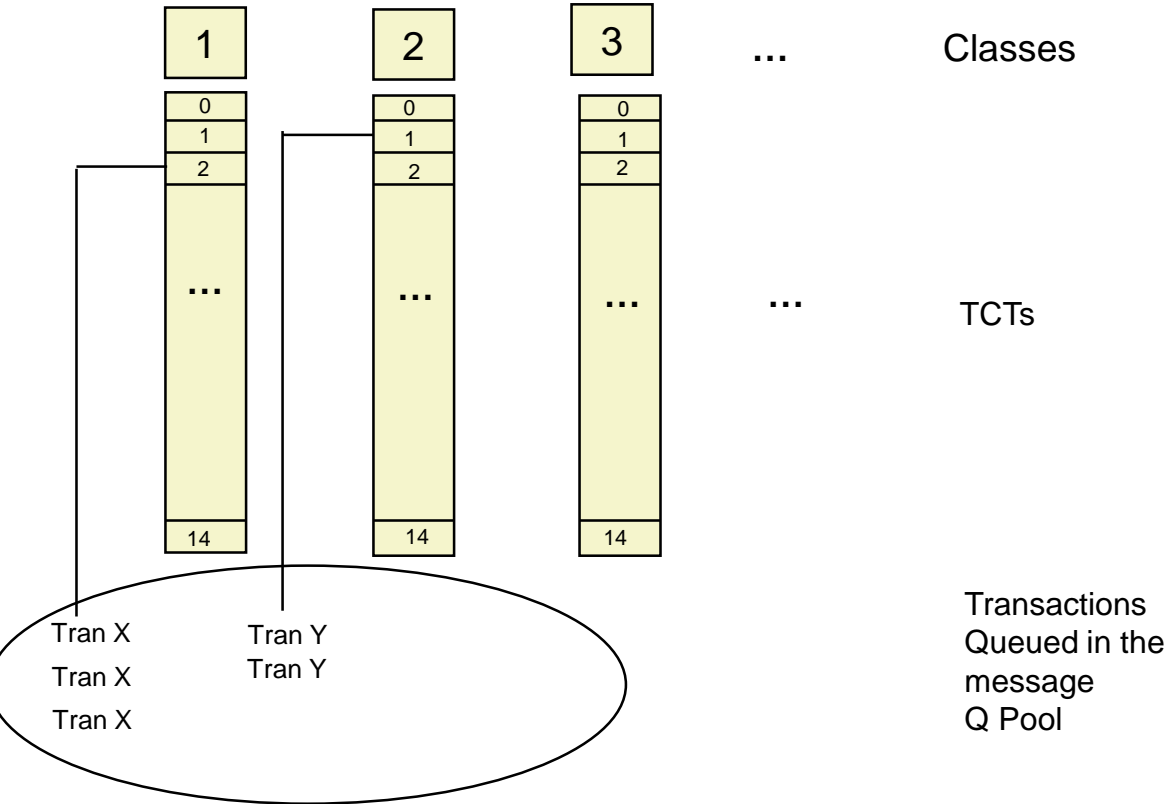
003



ORDER (6)
RECEIVE (4)

- Transactions are queued by Class and
 - Within class by priority
- Classes: 1 – 999
- Priority: 0 (Low) – 14 (High)

Transaction Queuing



Transaction Priorities

- Specified on the TRANSACT macro via PRTY parm:
 - PRTY (normal, Limit, Limit Count)

- Example:

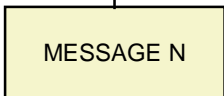
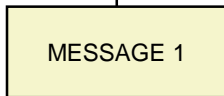
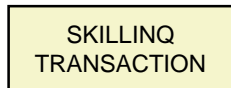
```
APPLCTN   PSB=PROG1, ..., PGMTYPE=(TP,,1)
          TRANSACT CODE=SKILLINQ, ..., PRTY=(5,10,30)
```

- Defaults:

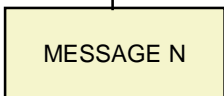
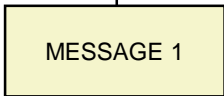
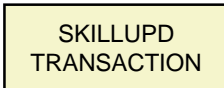
- Normal priority: 1
- Limit priority: 1
- Limit count: 65535

Transaction Priorities

CLASS 001



CLASS 001



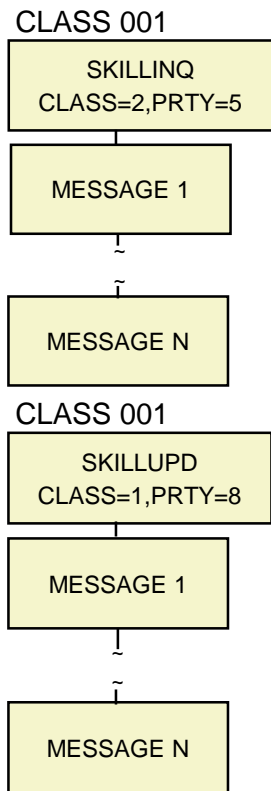
	TIME 1	TIME 2	TIME 3
--	-----------	-----------	-----------

NORMAL PRIORITY	5	5	5
LIMIT PRIORITY	10	10	10
LIMIT COUNT	30	30	30
QUEUE COUNT	8	31	0

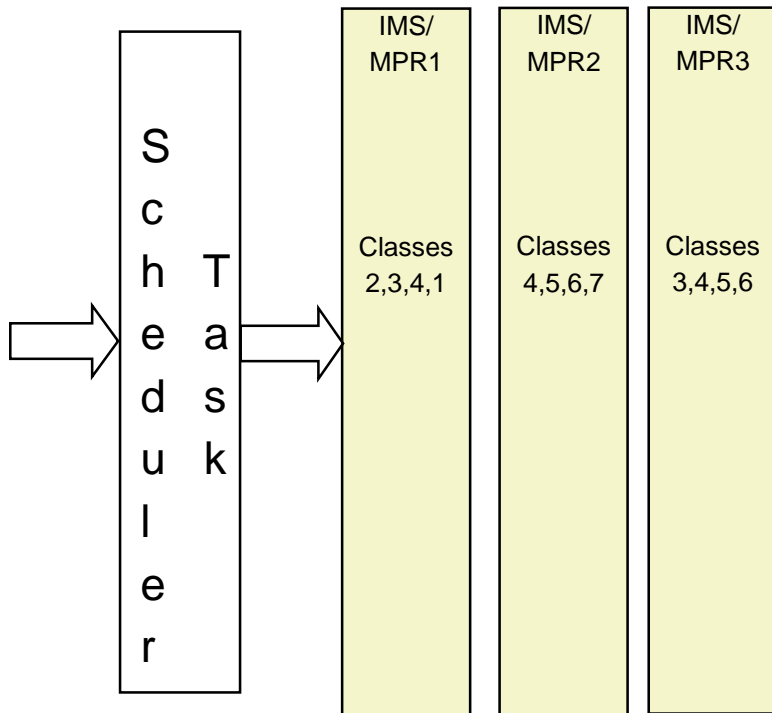
	TIME 1	TIME 2	TIME 3
--	-----------	-----------	-----------

NORMAL PRIORITY	8	8	8
LIMIT PRIORITY	12	12	12
LIMIT COUNT	4	4	4
QUEUE COUNT	2	1	3

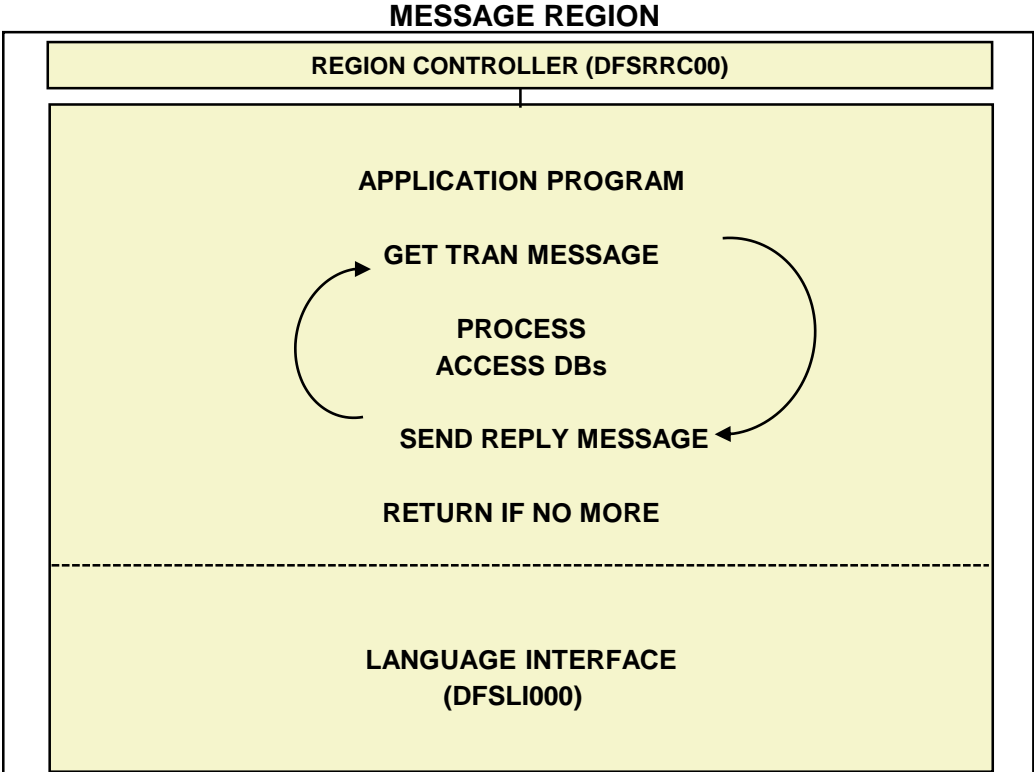
Transaction Scheduling



Application programs are automatically scheduled into Message Processing Regions (MPRs)



MPP Program flow



PROCLIM – Processing Limit Count

- Specified on the TRANSACT macro:

```
TRANSACT CODE=SKILLINQ, ..., PRTY=(5,10,30),  
                                PROCLIM=(count, time)
```

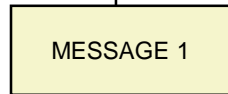
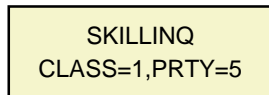
- Count: number of messages program can process in one scheduling.
 - Default: 65535 – no limit
- Time: CPU seconds to process one transaction

PROCLIM – Quick Reschedule

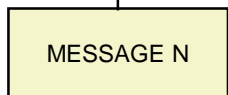
- Avoids unnecessary terminations & schedules.
- Invoked when PROCLIM>0
- When PROCLIM count reached
 - If same transaction would be rescheduled
 - Does not return 'QC' status code
 - Returns the next message
 - If same transaction would not be rescheduled
 - Returns 'QC' status code
 - Program terminates
- If PROCLIM count=0
 - Quick Reschedule is disabled
 - Only one transaction processed per schedule

WFI (Wait for Input)

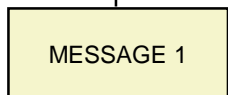
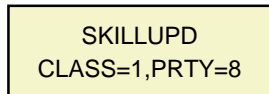
CLASS 001



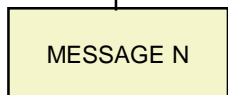
~



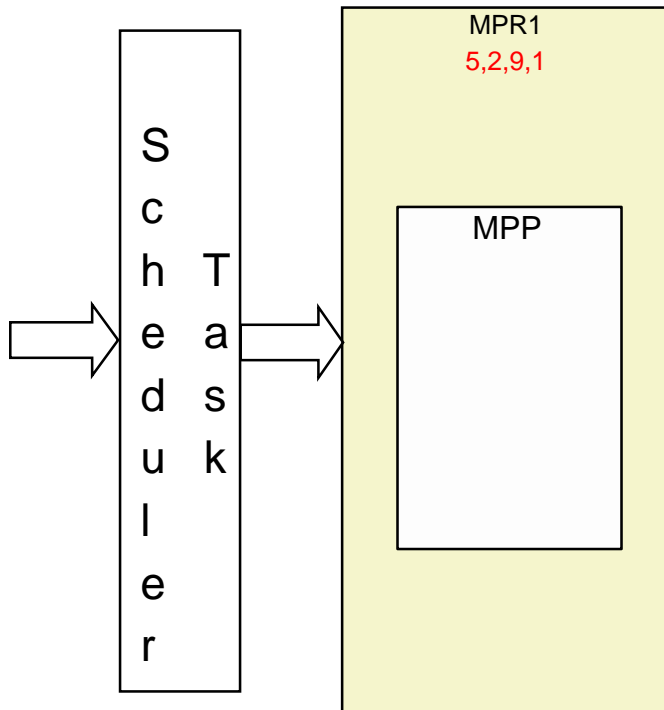
CLASS 001



~



Application programs are automatically scheduled into Message Processing Regions (MPRs)



WFI (Wait for Input) Transactions

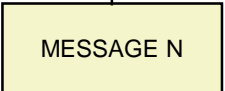
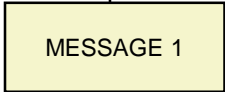
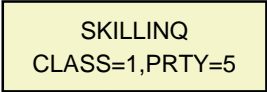
- Specified on the TRANSACT macro:
`TRANSACT CODE=SKILLINQ, ...,WFI, ...`
- Lack of message does not cause IMS to terminate program in region
 - GU to the IOPCB will NOT return 'QC' status code
 - Program is waiting for the next message to arrive
- Basically, provides a dedicated region for high priority transaction where response time is critical.
 - Avoids the scheduling overhead of those high priority transactions

WFI (Wait for Input) Transactions (cont.)

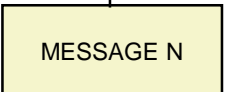
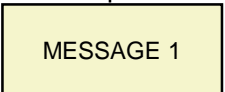
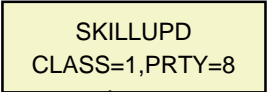
- WFI transactions do not give up their regions
- Programs terminated only by:
 - Reaching PROCLIM
 - /STOP TRANSACTION for MPP
 - /PSTOP Region reg# TRANSACTION tran for BMP
 - Shutdown checkpoint of IMS system
 - UPDATE PGM START(REFRESH) – IMS V14

PWFI (Pseudo Wait for Input)

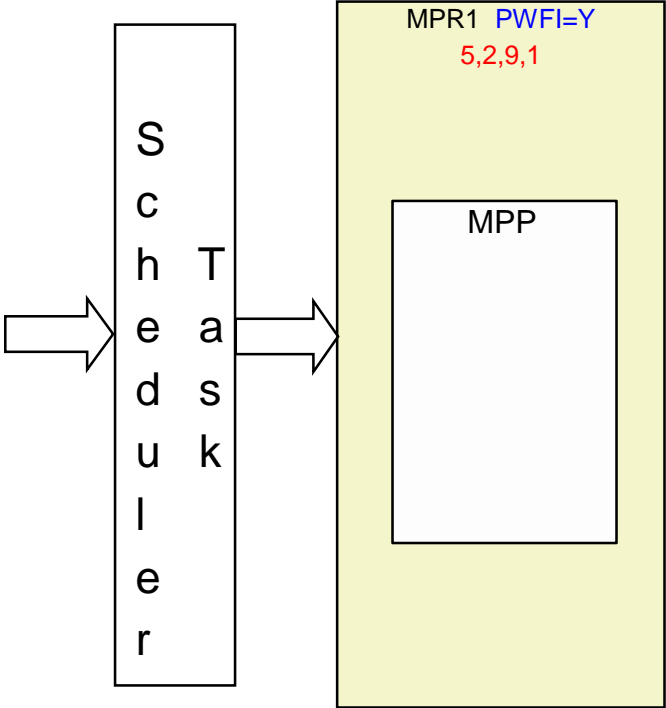
CLASS 001



CLASS 001



Application programs are automatically scheduled into Message Processing Regions (MPRs)



PWFI (Pseudo Wait for Input) Transactions

- Specified on the MPP region JCL:

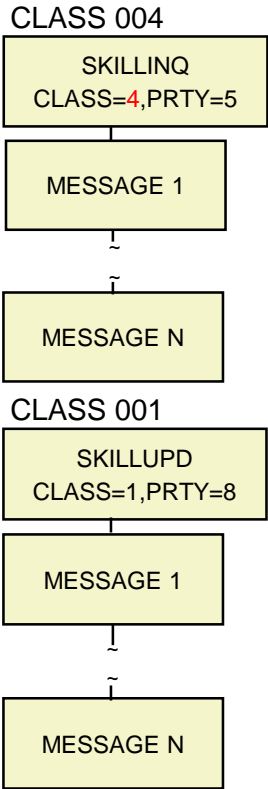
```
// EXEC PGM=DFSRRC00,PARM='MSG,....,PWFI=Y,....'
```

- Specifies what IMS will do when there are no more transaction messages to process:
 - GU to IOPCB will not result in 'QC' status code
 - Instead, waits for next message to arrive for the region
 - If next message is the same transaction code, great!! Avoids termination and scheduling of transaction.
 - If next message is different transaction code, returns 'QC', terminates program and schedules new program.

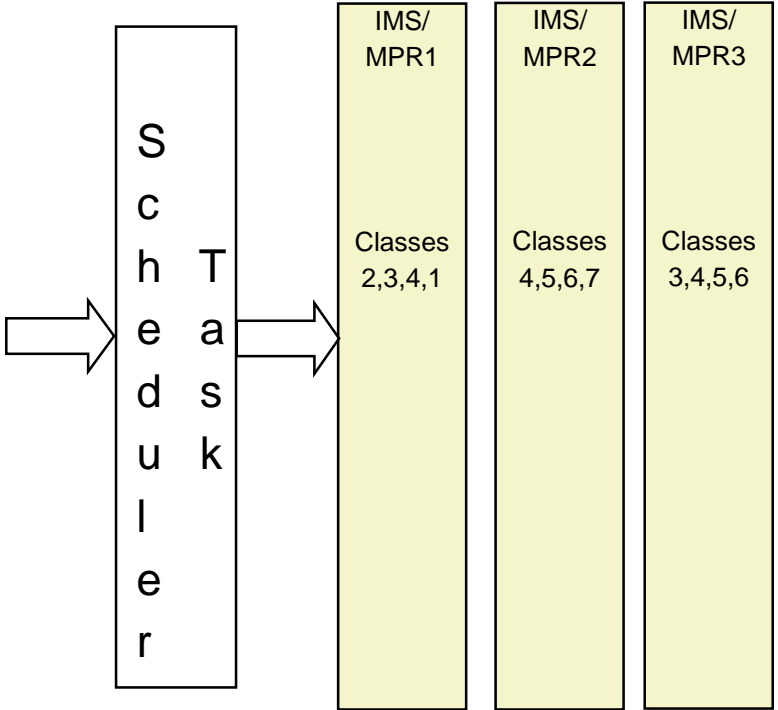
Dynamic Refresh for PWFI & WFI regions

- Introduced in IMS V14
- Via Type 2 command:
UPDATE PGM START(REFRESH)
- Regions do not need to be restarted.
- Programs get 'QC' status code if waiting for work or if processing a transaction, then after the current message is processed.

Serial vs Parallel Scheduling



Application programs are automatically scheduled into Message Processing Regions (MPRs)



PARLIM – Parallel Limit

- Specified on the TRANSACT Macro;
TRANSACT CODE=SKILLINQ, ..., **PARLIM=2**, ...
- Requires SCHDTYP=PARALLEL on APPLCTN macro
- Says transaction will be scheduled in another region if:

Tran Q Cnt > PARLIM x number of regions tran is currently scheduled

PARLIM – Parallel Limit (cont.)

- Example:

TRANSACT CODE=SKILL, ..., **PARLIM=2**, ...

SKILL Transaction current queue count=6

SKILL transaction currently scheduled in 2 MPRs.

3rd MPR becomes available

Can we schedule SKILL in the 3rd MPR?

PARLIM – Parallel Limit (cont.)

- A way to Load Balance the parallel scheduling of transactions
- $PARLIM=0$ always causes new transactions to be eligible for another available region

MAXRGN

- Specified on the TRANSACT Macro;
TRANSACT CODE=SKILLINQ, ..., MAXRGN=7, ...
- Limits the number of MPP regions that may have the transaction concurrently scheduled
- Used to keep a “dog” transaction from occupying too many regions

Unit Summary

- Queuing
- Scheduling
- Scheduling related parameters:
 - Class
 - MPR Class specification
 - Priority, Limit Priority & Limit Count
 - PROCLIM, Quick Reschedule
 - WFI, PWFI
 - Serial vs Parallel scheduling, PARLIM

Thank You!

