# Trace events - AddPexDfn's type(*trace) nnnEvt parameters

```
This file is based on the V4R2 info on addpexdfn's type(*trace) event parameters
lifted from the Performance Tools V4R2 book SC41-5340-00  Table 11-2
```

```
AddPexDfn type(*trace) sltevt(*yes) nnnEvt( )
          where nnnEvt = basEvt, pgmEvt, stgEvt etc
```

1. basevt( *prccrt, *prcdlt, *taskcrt, *pmco, *taskswtout,  etc )

2. pgmevt( *mistr, *miend, *mientry, *jvaentry, etc )
           These are also known as program bracketing events.

3. stgevt( *crtseg, *fndsegsiz, etc )

4. dskevt( *readstr, *pgreadstr, etc )

5. faultevt( *str, *endok, *enderr )

6. jobevt( *active, *ineligible, etc )

7. lckevt( *lwend )

8. sarevt( *clr, *read, *exchread, *wrt,  etc )

9. dsksvrevt( *ctlend, etc )

10. osevt( *mievnn thru *miev31 )

11. jvaevt( *objcrt, etc )

12. cmnevt( *service )

13. mchinst( list of MI's or *ALL )
             Need to specify pgmevt(*mistr *miend) too.

## Trace Event Descriptions:

```
The following tables describe each event available when you
specify SLTEVT(*YES) on the ADDPEXDFN command.

The tables also indicate the relationships between the TRCTYPE parameter
 and
the events that are included in the performance explorer definition.
```

### AddPexDfn BASEVT parameter

```
The data that is collected for the BASE events are placed in the
QAYPETIDX and QAYPEBASE files.
```

| Table 11-2. BASE event descriptions | AddPexDfn BASEVT parameter | |
|---|---|---|
| Event | TRCTYPE | Description |
| PRCCRT | | Process Create. An event is |

| | | |
|---|---|---|
| | | generated whenever an AS/400 process is created during an active performance explorer collection session. |
| PRCDLT | | Process Delete. An event is generated whenever an AS/400 process ends during an active performance explorer collection session. |
| TASKCRT | | Task Create. An event is generated whenever an AS/400 SLIC task is created during an active performance explorer collection session. |
| TASKDLT | | Task Delete. An event is generated whenever an AS/400 SLIC task ends during an active performance explorer collection session. |
| TASKSWTIN | *TASKSWT | Task Switch In. An event is generated whenever an AS/400 task is switched in and given processor time during an active performance explorer collection session. |
| TASKSWTOUT | *TASKSWT | Task Switch Out. An event is generated whenever an AS/400 task is switched out from the processor, due to time-slice end, during an active performance explorer collection session. |
| PMCO | ALL TRCTYPES | Performance Measurement Counter Overflow. An event is generated whenever this hardware counter overflows during an active performance explorer collection session. This hardware counter is used in conjunction with the Sample rate parameter and can be programmed to overflow and generate an event at the specified sample rate interval. |
| TASKSWTOUTQ | *TASKSWT | Task Switch Out Queueing. An event is generated whenever an AS/400 task is switched out from the processor, due to a lock condition, during an active performance explorer collection session. |
| TASKSWTOUTINT | *TASKSWT | Task Switch Out Interrup. An event is generated whenever an AS/400 task is switched out from the processor, due to an interrupt, during an active performance explorer collection session. |
| TASKAVAIL | *TASKSWT | Task Available for Dispatch. An event is generated whenever an AS/400 task is available to be dispatch to the processor during |

| | | |
|---|---|---|
| | | an active performance explorer collection session. |
| ACTGRPACTPGM | *BASIC, *PGMACT | Activation Group Activate Program. An event is generated whenever an ILE program activation begins during an active performance explorer collection session. |
| ACTGRPCRT | *BASIC, *PGMACT | Activation Group Create. An event is generated whenever an ILE program activation group is created during an active performance explorer collection session. |
| ACTGRPDLT | *BASIC, *PGMACT | Activation Group Delete. An event is generated whenever an ILE program activation group is destroyed during an active performance explorer collection session. |
| EXCP | | Exception. An event is generated whenever an exception occurs during an active performance explorer collection session. |
| MIEXCP | | Mi Exception. An event is generated whenever an MI exception occurs during an active performance explorer collection session. |
| EXTINTSTR | | External Interrupt Start. An event is generated whenever a start of an external interrrupt occurs during an active performance explorer collection session. |
| EXTINTEND | | External Interrupt End. An event is generated whenever an end of an external interrupt occurs during an active performance explorer collection session. |
| DCRINTSTR | | Decrement Interrupt Star. An event is generated whenever a start of a decrement interrupt occurs during an active performance explorer collection session. |
| DCRINTEND | | Decrement Interrupt End. An event is generated whenever an end of a decrement interrupt occurs during an active performance explorer collection session. |
| PGMDBG | *BASIC | Program Debug. This event is activated through the Dedicated Service Tools (DST) interface |
| SERVICE | | Service. Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

### [AddPexDfn](#) STGEVT parameter

Auxilary storage management (ASM) is responsible for the management of the
segment spaces for the single-level storage model of the AS/400.

For the non-heap (pre-V4R4) events,
   the data that is collected for the ASM events are placed in the
   QAYPETIDX and QAYPEASM files.


For the heap events,
the data that is collected for the ASM events are placed in the
QAYPETIDX and QAYPEHEAP files.

Table 11-3. Auxiliary storage management event descriptions
[AddPexDfn](#) STGEVT parameter

| Event | TRCTYPE | Description |
|-------|---------|-------------|
| CRTSEG | *BASIC, *DSKSTG, *VRTADR | Create Segment. An event is generated whenever a segment is created during an active performance explorer collection session. |
| DLTSEG | *BASIC, *DSKSTG | Destroy Segment. An event is generated whenever a segment is destroyed during an active performance explorer collection session. |
| EXDSEG | *BASIC, *DSKSTG | Extend Segment. An event is generated whenever a segment is extended during an active performance explorer collection session. |
| FNDSEGSIZ | *BASIC, *DSKSTG | Find Segment Size. An event is generated whenever a request to find the segment size occurs during an active performance explorer collection session. |
| TRUNCSEG | *BASIC, *DSKSTG | Truncate Segment. An event is generated whenver a segment is truncated during an active performance explorer collection session. |
| SYSHEAP | | System Heap. An event is generated whenever a system heap access is done by a task/job/thread in an active PEX collection session. |
| RESHEAP | | Resident Heap. An event is generated whenever a resident heap access is done by a task/job/thread in an active PEX collection session. |
| LCLHEAP | | Local Heap. An event is generated whenever a local heap |

| | | |
|---|---|---|
| | | access is done by a task/job/thread in an active PEX collection session. |
| USRHEAP | | User Heap. An event is generated whenever a user heap access is done by a task/job/thread in an active PEX collection session. |
| ACTGRPHEAP | | Activation Group Heap. An event is generated whenever an activation group heap access is done by a task/job/thread in an active PEX collection session. |
| HDLHEAP | | Handle Based Heap. An event is generated whenever a handle based heap access is done by a task/job/thread in an active PEX collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## **AddPexDfn** DSKEVT parameter

Disk events trace I/O activity with the disk storages devices of the AS/400.

The data that is collected for the Disk events are placed in the QAYPETIDX and QAYPEDASD files.

Table 11-4. Disk event descriptions    AddPexDfn DSKEVT parameter

| Event | TRCTYPE | Description |
|---|---|---|
| READSTR | *BASIC, *DSKIO1, *DSKIO2, *TASKSWT | Read Start. An event is generated whenever a start of a physical read occurs during an active performance explorer collection session. |
| READEND | *DSKIO1, *DSKIO2, *TASKSWT | Read End. An event is generated whenever a complete of a physical read occurs during an active performance explorer collection session. |
| WRTSTR | *BASIC, *DSKIO1, *DSKIO2, *TASKSWT | Write Start. An event is generated whenever a start of a physical write occurs during an active performance explorer collection session. |
| WRTEND | *DSKIO1, *DSKIO2, *TASKSWT | Write End. An event is generated whenever a complete of a physical write occurs during an active performance explorer collection session. |
| PGREADSTR | | Page Read Start. An event is |

| | | generated whenever a start of a page read occurs during an active performance explorer collection session. |
| PGREADEND | | Page Read End. An event is generated whenever a complete of a page read occurs during an active performance explorer collection session. |
| PGWRTSTR | | Page Write Start. An event is generated whenever a start of page write occurs during an active performance explorer collection session. |
| PGWRTEND | | Page Write End. An event is generated whenever a complete of a page write occurs during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## [AddPexDfn](#) FAULTEVT parameter

A page fault occurs whenever an address is referred to and is not in the
main storage of the AS/400 system.

The data that is collected for the page fault events are placed in the
QAYPETIDX and QAYPEPGFLT files.

| Table 11-5. Page fault event descriptions  [AddPexDfn](#) FAULTEVT parameter | | |
|---|---|---|
| Event | TRCTYPE | Description |
| STR | *BASIC, *DSKIO2 | Start of Fault. An event is generated whenever a start of a page fault occurs during an active performance explorer collection session. |
| ENDOK | | End of Fault: OK. An event is generated whenever a successful completion of a page fault occurs during an active performance explorer collection session. |
| ENDERR | | End of Fault: Error. An event is generated whenever an unsuccesful completion of a page fault occurs during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## [AddPexDfn](#) JOBEVT parameter

The mangagement of the MI interrupts are for the jobs or processes running
on the AS/400 system

The data that is collected for the RMPM events are placed in the
QAYPETIDX and QAYPERMPM files.

Table 11-6. Job or Process  Events
         [AddPexDfn](#) JOBEVT parameter

| Event | TRCTYPE | Description |
|---|---|---|
| LWSTR | | Machine Interface Long Wait Start. An event is generated whenever a start of a long wait occurs during an active performance explorer collection session. |
| LWEND | | Machine Interface Long Wait End. An event is generated whenever a completion of a long wait occurs during an active performance explorer collection session. |
| INELIGIBLE | | Machine Interface Process Ineligible. An event is generated whenever a process moves from active state to ineligible during an active performance explorer collection session. |
| ACTIVE | | Machine Interface Process Active. An event is generated whenever a process moves from ineligible or wait to active during an active performance explorer collection session. |
| INTERRUPT | | Machine Interface Process Interrupt. An event is generated whenever a process interrupt occurs during an active performance explorer collection session. |
| MPLPOOLCHG | | Machine Interface MPL Pool Change. An event is generated whenever the MPL pool changes during an active performance explorer collection session. |
| TOBCHMPLPOOL | | Machine Interface Process To Batch MPL Pool. An event is generated whenever an MI process is being implicitly moved to the batch MPL pool defined for it during an active performance explorer collection session. |
| TSLEND | | Machine Interface Timeslice End. An event is generated whenever a process timeslice end occurs during an active performance |

| | | explorer collection session. |
|---|---|---|
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## **AddPexDfn** **LCKEVT parameter**

The mangagement of the seizes and locks used for resource serialization
that occurr on the AS/400 system.

The data that is collected for the RMSL events are placed in the
QAYPETIDX and QAYPERMSL files.

Table 11-7. Seize Lock Events  **AddPexDfn** LCKEVT parameter

| Event | TRCTYPE | Description |
|---|---|---|
| LWEND | | Sieze Lock Long Seize Wait End. An event is generated whenever a long seize wait completes during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## **AddPexDfn** **SAREVT parameter**

The SAR events (give description here)

The data that is collected for the SAR events are placed in the
QAYPETIDX and QAYPESAR files.

Table 11-8. Segment Address Register Events
        **AddPexDfn** SAREVT parameter

| Event | TRCTYPE | Description |
|---|---|---|
| CLR | *BASIC | Clear. An event is generated whenever a storage clear operation is requested during an active performance explorer collection session. |
| CLRPIN | *BASIC | Clear Pin. An event is generated whenever a storage clear and pin operation is requested during an active performance explorer collection session. |
| EXCHCLR | *BASIC | Exchange Clear. An event is generated whenever a storage exchange and clear operation is requested during an active |

| | | performance explorer collection session. |
|---|---|---|
| READ | *BASIC, *DSKIO2 | Read. An event is generated whenever a storage read operartion is requested during an active performance explorer collection session. |
| READASYNC | *BASIC, *DSKIO2 | Read Asynchronous. An event is generated whenever a storage asynchronous read operartion is requested during an active performance explorer collection session. |
| READASYNCCALL | *BASIC, *DSKIO2 | Read Asynchronous Track Call. An event is generated whenever the storage manangement read asynchronous tracked by a call function is accessed during an active performance explorer collection session. |
| READPIN | *BASIC, *DSKIO2 | Read Pin. An event is generated whenever a storage read and pin operation is requested during an active performance explorer collection session. |
| EXCHREAD | *BASIC, *DSKIO2, *FILEOPEN | Exchange Read. An event is generated whenever a storage exchange read operation is requested during an active performance explorer collection session. |
| EXCHREADASYNC | *BASIC, *DSKIO2 | Exchange Read Asynchronous. An event is generated whenever a storage asynchronous exchange read operation is requested during an active performance explorer collection session. |
| EXCHREAD- ASYNCCALL | *BASIC, *DSKIO2 | Exchange Read Asynchronous Track Call. An event is generated whenever the storage manangement exchange read asynchronous tracked by a call function is accessed during an active performance explorer collection session. |
| WRT | *BASIC, *DSKIO2 | Write. An event is generated whenever a storage write operation is requested during an active performance explorer collection session. |
| WRTASYNC | *BASIC, *DSKIO2 | Write Asynchronous. An event is generated whenever a storage asynchronous write operation is requested during an active performance explorer collection session. |
| WRTASYNCCALL | *BASIC, *DSKIO2 | Write Asynchronous Track Call. An event is generated whenever the storage manangement write |

| | | |
|---|---|---|
| | | asynchronous tracked by a call function is accessed during an active performance explorer collection session. |
| WRTASYNCMSGQ | *BASIC, *DSKIO2 | Write Asynchronous Track Message Queue. An event is generated whenever the storage manangement write asynchronous function is accessed during an active performance explorer collection session. |
| WRTASYNC-MSGQLAST | *BASIC, *DSKIO2 | Write Asynchronous Track Message QueueLast. An event is generated whenever the storage manangement write last asynchronous tracked by a call is requested during an active performance explorer collection session. |
| WRTASYNCTASK | *DSKIO2 | Write Asynchronous Track Task. An event is generated whenever the storage manangement write last asynchronous tracked by the current task is requested during an active performance explorer collection session. |
| WRTASYNCWAIT | | Write Asynchronous Track Wait. An event is generated whenever the storage manangement write asynchronous tracked by is requested during an active performance explorer collection session. |
| WRTRMV | *BASIC, *DSKIO2 | Write Remove. An event is generated whenever a storage write remove operation is requested during an active performance explorer collection session. |
| WRTPGOUT | *BASIC, *DSKIO2 | Write For Page Out. An event is generated whenever a storage write for page out operation is requested during an active performance explorer collection session. |
| RMV | *BASIC | Remove. An event is generated whenever a storage remove operation is requested during an active performance explorer collection session. |
| UNPIN | *BASIC | Unpin. An event is generated whenever a storage unpin operation is requested during an active performance explorer collection session. |
| UNPINRMV | *BASIC | Unpin Remove. An event is generated whenever a storage unpin remove operation is requested during an active performance explorer collection session. |

| Event | TRCTYPE | Description |
|---|---|---|
| UNPINWRT | *BASIC, *DSKIO2 | Unpin Write. An event is generated whenever a storage unpin write operation is requested during an active performance explorer collection session. |
| ENDOK | | End of SAR event: OK. An event is generated whenever any specified SAR event completes successfully during an active performance explorer collection session. |
| ENDERR | | End of SAR event: Error. An event is generated whenever any specified SAR event completes with an error during an active performance explorer collection session. |
| STRASYNCTASK | | Start Asynchronous Task. An event is generated whenever during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## [AddPexDfn](#) DSKSVREVT parameter

The DASD server events (give descriptions here)

The data that is collected for the DASD server events are placed in the QAYPETIDX and QAYPEDSRV files.

| Table 11-9. DASD Server Events [AddPexDfn](#) DSKSVREVT parameter | | |
|---|---|---|
| Event | TRCTYPE | Description |
| CTLEND | *DSKSVR | Control End. An event is generated whenever a server request to create or destroy session completes during an active performance explorer collection session. |
| RQSACP | *DSKSVR | Request Accepted. An event is generated whenever a server function request is accepted and will be handled by a server task during an active performance explorer collection session. |
| RQSRJT | *DSKSVR | Request Rejected. An event is generated whenever a server function request is rejected because of an error during an active performance explorer collection session. |

| Event | TRCTYPE | Description |
|---|---|---|
| SVRTASKSTR | *DSKSVR | Server Task Started. An event is generated whenever a server task is created and is ready to process requests during an active performance explorer collection session. |
| SVRTASKEND | *DSKSVR | Server Task Ended. An event is generated whenever a server task is ended and will be removed from the system during an active performance explorer collection session. |
| RQSRCV | *DSKSVR | Request Received. An event is generated whenever a server task receives a request for work (read or write) during an active performance explorer collection session. |
| RQSENDOK | *DSKSVR | Request Ended OK. An event is generated whenever a server task completes the request with no errors during an active performance explorer collection session. |
| RQSENDERR | *DSKSVR | Request Ended Error. An event is generated whenever a server task encounters an error while processing a request during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## [AddPexDfn](#) PGMEVT parameter

Program bracket events trace the call flow of programs and procedures as
well as MI complex instructions. These events give information on when
programs, procedures and MI complex instruction are entered into (or
called) and exited from (or returned).

The data that is collected for the program/procedure bracketing events are placed in the
QAYPETIDX and QAYPEMBRKT files.

Table 11-10. Program bracket events
[AddPexDfn](#) PGMEVT parameter

| Event | TRCTYPE | Description |
|---|---|---|
| MIENTRY | *CALLRTN, *FILEOPEN, *PGMACT | Entry. An event is generated whenever a program or procedure that is enabled for performance collection is entered or called during an active performance explorer collection session. |
| MIEXIT | *CALLRTN, | Exit. An event is generated |

| | | |
|---|---|---|
| | *FILEOPEN, *PGMACT | whenever a program or procedure that is enabled for performance collection exits or returns during an active performance explorer collection session. |
| MISTR | *CALLRTN, *PGMACT | Machine Interface Instruction Start. An event is generated whenever an MIcomplex instruction is routed to or from a program during an active performance explorer collection session. |
| MIEND | *CALLRTN, *PGMACT | Machine Interface Instruction End. An event is generated whenever an MI complex instruction completes and returns to the program during an active performance explorer collection session. |
| MIPRECALL | | Pre Call. An event is generated whenever a program or procedure that is enabled for performance collection calls externally to another program or procedure during an active performance explorer collection session. |
| MIPOSTCALL | | Post Call. An event is generated whenever a program or procedure that is enabled for performance collection has been returned to from an external call to another program or procedure during an active performance explorer collection session. |
| JVAENTRY | | Java Entry. An event is generated whenever a Java method is called during an active performance explorer collection session. |
| JVAEXIT | | Java Exit. An event is generated whenever a Java method is returned from during an active performance explorer collection session. |
| JVAPRECALL | | Java Precall. An event is generated whenever a Java method is about to call out some other function during an active performance explorer collection session. |
| JVAPOSTCALL | | Java Postcall. An event is generated whenever a Java method is returned from some other function during an active performance explorer collection session. |
| JVANTVMTHSTR | | Java Native Method Start. An event is generated whenever a Java method calls an AS/400 |

| | | |
|---|---|---|
| | | method during an active performance explorer collection session. |
| JVANTVMTHEND | | Java Native Method End. An event is generated whenever a Java method returns from an AS/400 method during an active performance explorer collection session. |

## **AddPexDfn** OSEVT parameter

OS events describe activity occurring in various 'above the MI' components
of the AS/400 system.

The data that is collected for the OS events is placed in the
QAYPETIDX and QAYPEMIUSR files.

The V4R4 AddPexDfn command supports the OSEVT parameter.

-------------------------------------------------------------------------

| Event | TRCTYPE | Description |
|---|---|---|
| DBIO | | Database IO. An event is generated whenever a database logical data i/o is done by a job/thread in an active PEX collection session. |
| DBOPEN | | Database Open.  An event is generated whenever a database full file open or close is done by a job/thread in an active PEX collection session. |
| DTAARA | | Data Area. An event is generated whenever a data area (*dtaara) access is done by a job/thread in an active PEX collection session. |
| DTAQ | | Data Queue. An event is generated whenever a data queue (*dtaq) access is done by a job/thread in an active PEX collection session. |

## **AddPexDfn** JVAEVT parameter

Java events describe activity occurring in the Java virtual machine of the
AS/400 system.

The data that is collected for the Java events are placed in the
QAYPETIDX and QAYPEJVA files.

Table 11-11. Java events
          AddPexDfn JVAEVT parameter

| Event | TRCTYPE | Description |
|-------|---------|-------------|
| OBJCRT | | Java object create. An event is generated whenever a Java object instance is created during an active performance explorer collection session. |
| GBGCOL | | Java garbage collection. An event is generated whenever Java is doing garbage collection during an active performance explorer collection session. |
| GBGCOLSWEEP | | Java garbage collection sweep. An event is generated whenever the garbage collection sweep completes during an active performance explorer collection session. |
| THDCRT | | Java thread create. An event is generated whenever a Java thread is created during an active performance explorer collection session. |
| THDDLT | | Java thread delete. An event is generated whenever a Java thread is destructed during an active performance explorer collection session. |
| THDSSP | | Java thread suspend. An event is generated whenever a Java thread or thread group suspend is started during an active performance explorer collection session. |
| THDRSM | | Java thread resume - An event is generated whenever a java thread or thread group is resumed during an active performance explorer collection session. |
| THDWAIT | | Java thread wai. An event is generated whenever an Object::wait() used for thread synchronization is started during an active performance explorer collection session. |
| THDNFY | | Java thread notify. An event is generated whenever an Object::notify() used for thread synchronization is started during an active performance explorer collection session. |
| THDNFYALL | | Java thread notify all. An event is generated whenever an Object::notifyAll() used for thread synchronization is started during an active performance explorer collection session. |

| | | |
|---|---|---|
| THDSTTCHG | | Java thread state change. An event is generated whenever a method that causes a thread state transition (for example, a thread block) is started during an active performance explorer collection session. |
| CLSLOAD | | Java class load. An event is generated whenever a Java class is loaded during an active performance explorer collection session. |
| CLSUNLOAD | | Java class unload. An event is generated whenever a Java class is unloaded during an active performance explorer collection session. |
| LIBOPR | | Java class library operation. An event is generated whenever a Java native method library is loaded or unloaded during an active performance explorer collection session. |
| TFMSTR | | Java transform start. An event is generated whenever a Java program object generation step is started during an active performance explorer collection session. |
| LCKSTR | | Java lock. An event is generated whenever a synchronize lock is requested during an active performance explorer collection session. |
| UNLCK | | Java unlock. An event is generated whenever a synchronize lock is released during an active performance explorer collection session. |
| JVAEND | | Java end. An event is generated whenever a previous Java event is completed during an active performance explorer collection session. |
| SERVICE | | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## **AddPexDfn CMNEVT parameter**

Communication events describe activity occurring in the TCP/IP related communications component(s) of the AS/400 system.

The data that is collected for the communication events is placed in the QAYPETIDX and QAYPEMIUSR files.

|                          Communication events                          |||
|        AddPexDfn CMNEVT parameter                                       |||
| Event | TRCTYPE | Description |
|---|---|---|
| SERVICE |  | Service.  Use this value when your Service Rep requests it. Event's data is in qaypeusrdf. |

## AddPexDfn MCHINST parameter

| Table 11-12. Selected MI complex instructions |||
|        AddPexDfn MCHINST parameter                                      |||
| MI INST | TRCTYPE | Description |
|---|---|---|
| *REQIO | *BASIC | Request IO complex MI |
| *SNSEXCPD | *BASIC | Sense exception description |
| *DEQ | *BASIC | Dequeue |
| *ENQ | *BASIC | Enqueue |
| *WAITEVT | *BASIC | Wait on event |
| *RETEVTD | *BASIC | Retrieve event data |
| *SUSPR | *BASIC | Suspend process |
| *TERMPR | *BASIC | Terminate process |
| *INITPR | *BASIC | Initiate process |
| *RESPR | *BASIC | Resume process |
| *SIGEVT | *BASIC | Signal event |
| *UNLOCKSL | *BASIC | Unlock space location |
| *LOCK | *BASIC | Lock object |
| *LOCKSL | *BASIC | Lock space location |
| *UNLOCK | *BASIC | Unlock object |
| *DEQWAIT | *BASIC | Dequeue wait |
| *DEACTPG | *BASIC, *PGMACT | De-activate program |
| *DESAGP | *BASIC, *PGMACT | Destroy activation group |