

IBM

@server

iSeries

Independent disk pools





@server

iSeries

Independent disk pools

Contents

Independent disk pools	1
What's new for V5R2	2
Print this topic	2
Independent disk pools concepts	2
Independent disk pool terminology.	3
How independent disk pools work	5
Switchable and standalone independent disk pools	6
Benefits of independent disk pools	7
Types of disk pools	8
Disk pool groups	9
Recommended structure for independent disk pools.	10
Independent disk pool restrictions and considerations	11
Plan for independent disk pools	15
Hardware requirements	15
Physical planning requirements	16
Software and licensing requirements	17
Communications requirements.	17
Cluster requirements	17
Application considerations for independent disk pools	18
Configure independent disk pools	19
Access disk management functions	19
Create a switchable independent disk pool	20
Create a standalone independent disk pool	25
Create a new disk pool group	25
Convert UDFS disk pools	26
Manage independent disk pools	26
Make a disk pool unavailable	27
Recover an independent disk pool	27
Switch access to backup server	28
Change the server takeover IP address	28
Backup and recovery of independent disk pools	28
Examples: Independent disk pools configurations.	29
Standalone independent disk pools	29
Switchable independent disk pools	30
Frequently asked questions.	33
Warning: Temporary Level 3 Header	34
Related information.	37

Independent disk pools

The terms **independent auxiliary storage pool (ASP)** and **independent disk pool** are synonymous.

An independent disk pool is a collection of disk units that can be brought online or taken offline independent of the rest of the storage on a system, including the system disk pool, basic user disk pools, and other independent disk pools. An independent disk pool can be either:

- switchable among multiple systems in a clustered environment, or
- privately connected to a single system.

The benefits, in both multi-system clustered environments and single-system environments, can be significant. For example, in a clustered environment, the use of independent disk pools can provide disk storage that is switchable amongst servers in the cluster, providing continuous availability of resources. In a single-system environment, independent disk pools could be used to isolate infrequently used data that does not always need to be present when the system is operational.

This topic will provide you with the information you need to implement independent disk pools, from a conceptual explanation to planning, configuring, and managing independent disk pools on your servers.

What's new for V5R2

Take a look at what is new for this release.

Print this topic

View or download a PDF version of this Independent disk pools topic for viewing or printing.

Independent disk pools concepts

Learn about how independent disk pools work, as well as their benefits and uses.

Plan for independent disk pools

Depending upon how you plan to use independent disk pools, there are hardware, software, and communications requirements that must be met. Use this information to identify prerequisites for your desired implementation.

Application considerations for independent disk pools

If you write applications for an independent disk pool environment, you should be aware of these unique considerations.

Configure independent disk pools

Read about how iSeries Navigator helps you configure independent disk pools.

Manage independent disk pools

After you have your independent disk pools created and configured, use this topic to understand how to manage them.

Examples: Independent disk pools configurations

Whether you are implementing in a single-system or a multi-system clustered environment, see some examples of how independent disk pools can be used.

Frequently asked questions (FAQ)

See some frequently asked independent disk pool questions and answers.

Related information

IBM^(R) related information contains technical, know-how, and "how to" information.

What's new for V5R2

Independent disk pools provide the ability to group together storage that can be taken offline or brought online independent of system data or other unrelated data. Independent disk pools have been enhanced in V5R2 to provide support for:

- **Library-based objects**

When independent disk pools were introduced in V5R1, they supported user-defined file systems (UDFS) only. Support for library-based objects has been added in V5R2. See Supported and unsupported OS/400 object types for details.

- **Up to 223 independent disk pools**

You can now create as many as 223 independent disk pools. Previous releases only supported 67 independent disk pools. In V5R1 independent disk pools were numbered from 33-99. That range has been expanded to 33-255 at V5R2.

- **Disk pool groups**

A disk pool group is made up of a primary disk pool and zero or more secondary disk pools, each of which are independent in regard to data storage, but combine to act as one entity. See Disk pool groups.

- **Multiple databases**

When an independent disk pool is created, it will appear as a distinct user database on the server. This is separate from the system database, which was the only database available per system in previous releases. See Independent disk pools with distinct databases.

To find other information about what's new or changed this release, see the Memo to Users  .


Print this topic

To view or download the PDF version of this topic, select Independent disk pools (about 360 KB or 44 pages).

Saving PDF files

1. Open the PDF in your browser (click the link above).
2. In the menu of your browser, click **File**.
3. Click **Save As...**
4. Navigate to the directory in which you would like to save the PDF.
5. Click **Save**.

Downloading Adobe Acrobat Reader

If you need Adobe Acrobat Reader to view or print these PDFs, you can download a copy from the Adobe Web site (www.adobe.com/prodindex/acrobat/readstep.html)  .

Independent disk pools concepts

Before you implement independent disk pools in your environment, it is important to understand some key concepts, including important terminology, as well as how independent disk pools work and how they can be beneficial.

See the following topics for a conceptual understanding of independent disk pools:

- Terminology

- How independent disk pools work
- Switchable and standalone independent disk pools
- Benefits of independent disk pools
- Types of disk pools
- Contrasting basic and independent disk pools
- Disk pool groups
- Restrictions and considerations
- Recommended structure for independent disk pools

Independent disk pool terminology

As you work with independent disk pools, you will need to become familiar with the following terms. For more terms and concepts, you can access the Information Center glossary.

Important: The terms **independent auxiliary storage pool (ASP)** and **independent disk pool** are synonymous.

cluster

A collection of complete systems that work together to provide a single, unified computing capability. An iSeries cluster is made up of only iSeries servers and is required when implementing switchable independent disk pools.

cluster resource group (CRG)

A collection of related cluster resources that defines actions to be taken during a switchover or failover operation of the access point of resilient resources. The group describes a recovery domain and supplies the name of the cluster resource group exit program that manages the movement of an access point. A device CRG contains a list of switchable devices, such as independent disk pools which reside on a switchable entity. A switchable entity can be either an expansion unit (tower) or an IOP. In iSeries Navigator, a device cluster resource group is referred to as a switchable hardware group.

device description

An object that contains information describing a particular device or logical unit (LU) that is attached to the system. A device description is a description of the logical connection between two LUs (local and remote locations). The system-recognized identifier for the object type is *DEVD.

device domain

A device domain is a collection of cluster nodes that share device resources, such as independent disk pools. For independent disk pools, the resources are: virtual addresses, disk pool numbers and disk unit numbers. An independent disk pool can only be accessed by the nodes in one device domain.

disk pool

An auxiliary storage pool that contains only disk units. See Types of disk pools.

disk pool group

Made up of a primary disk pool and zero or more secondary disk pools, each of which are independent in regard to data storage, but combine to act as one entity. See Disk pool groups.

disk unit

A physical enclosure containing one or more disk drives.

expansion unit

A feature that can be connected to a system unit to provide additional storage and processing capacity. Synonymous with *tower*.

failover

A cluster event where the primary database server or application server switches over to a backup system due to the failure of the primary server

HSL (high-speed link) loop

The system-to-tower connectivity technology that is required to implement switchable independent disk pools residing on an expansion unit (tower). The servers and towers in a cluster using resilient devices on an external tower must be on and HSL loop connecting with HSL cables.

independent disk pool

One or more storage units that are defined from the disk units or disk-unit subsystems that make up addressable disk storage. An independent disk pool contains objects, the directories and libraries that contain the objects, and other object attributes such as authorization ownership attributes. An independent disk pool can be made available (varied on) and made unavailable (varied off) without restarting the system. An independent disk pool can be either a) switchable among multiple systems in a clustering environment or b) privately connected to a single system. Synonymous with *independent auxiliary storage pool (ASP)*.

library name space

An attribute that can be set for the current thread. The library name space is the set of objects and libraries that can be accessed in any independent disk pools in a disk pool group plus the libraries in the system disk pool and basic user disk pools (ASPs 2-32) using the regular library-qualified object name syntax. The Set Auxiliary Storage Pool Group (SETASPGRP) command sets the auxiliary storage pool (ASP) group for the current thread.

primary disk pool

An independent disk pool that defines a collection of directories and libraries and may have other secondary disk pools associated with it. A primary disk pool also defines a database for itself and other disk pools that may be added in its disk pool group. Primary disk pools can only be implemented on V5R2 or later of OS/400. See Types of disk pools.

secondary disk pool

An independent disk pool that defines a collection of directories and libraries and must be associated with a primary disk pool. Secondary disk pools can only be implemented on V5R2 or later of OS/400. See Types of disk pools.

switchable entity

The physical resource containing the independent disk pools that can be switched between systems in a cluster. This can be an expansion unit containing disk units in a multiple system environment. This could also be an IOP containing disk units in an LPAR environment.

switchover

A cluster event where the primary database server or application server switches over to a backup system due to the manual intervention from the cluster management interface.

SYSBAS

In the character-based interface, refers to the system ASP (ASP 1) and all configured basic ASPs (ASPs 2-32). Independent disk pools (ASPs 33-255) are not included.

UDFS disk pool

An independent disk pool that contains only user-defined file systems. It cannot be a member of a disk pool group unless it is converted to a primary or secondary disk pool. See Types of disk pools.

vary off

To make an independent disk pool unavailable for its normal, intended use. All of the primary and secondary disk pools in a disk pool group will vary off together. Synonymous with *make unavailable*.

vary on

To make an independent disk pool available for its normal, intended use. All of the primary and secondary disk pools in a disk pool group will vary on together. Synonymous with *make available*.

How independent disk pools work

The key characteristic of an independent disk pool is its ability to be, of course, *independent* of the rest of the storage on a server. It is independent because the data in the independent disk pool is self-contained. This means that all of the necessary system information associated with the data resides within the independent disk pool. The unique qualities of an independent disk pool allow it to be switched in a multi-system environment and to be made available and unavailable in a single-system environment.

Independent disk pools are available only when you choose to make them available; they are not made available during a normal restart of your server, unless you include code in your Start Up Program to make them available. When you select to make a disk pool available, the disk pool goes through a process similar to that of a server restart. While this processing takes place, the disk pool is in an Active state.

While the disk pool is in Active state, recovery steps are being performed. The disk pool is synchronized with other disk pools that may be in the disk pool group. Also, journaled objects are synchronized with their associated journal. System libraries are created for the primary disk pool: QSYSnnnnn, QSYS2nnnnn, QRCLnnnnn, QRCYnnnnn, QRPLnnnnn, SYSIBnnnnn (where *nnnnn* is the primary disk pool number right justified and padded with zeroes). For example, the QSYS library for independent disk pool 33 is QSYS00033.

At this time database cross-reference files will also be updated. The system libraries for the independent disk pool QSYSnnnnn and QSYS2nnnnn contain metadata not only for the independent disk pool, but also for the system disk pool. When the disk pool is made available, database cross-referencing clears the information related to SYSBAS and updates it with current information. The number and complexity of database file objects and SQL packages, procedures, and functions that need to be updated will play a role in the time it takes to make the disk pool available.

During the make available process, several server jobs are started to support the independent disk pool. In order for server jobs to remain unique on the server, those that service the independent disk pool are given their own simple job name when the disk pool is made available. The server jobs are essential to the operation of the disk pool; do not tamper with these server jobs. The following is a list of server jobs that are created to run in the QSYSWRK subsystem:

1. **QDBXnnnXR** - handles database cross-reference file server functions
2. **QDBXnnnXR2** - handles database cross-reference field (column) information
3. **QDBnnnSV01** - handles database, journal and commitment control events
4. **QDBnnnSV02 through QDBnnnSV##** - high priority jobs that service the database
5. **QDBnnnSV## through QDBnnnSV##** - low priority jobs that service the database

When the recovery process completes, the disk pool is in an Available state, ready for you to use. When you make a disk pool group available, you will see a completion messages for each disk pool. If the make available process encounters problems, such as an object not synchronized with a journal, you will need to address the issues reported in the error messages. See the job log, the system operator message queue, and the history log to locate problems and verify the make available process.

Example: Make independent disk pool available at startup

If you would like your independent disk pools to be made available in most cases when your server is restarted, you may want to consider including the following control language (CL) at the beginning of your Start Up Program (QSTRUP). When you do not want the independent disk pools to be made available when you restart the server, you can either Delete the Data Area (DLTDTAARA) or Rename it (RNMOBJ). However, you will need to remember to either Create the Data Area (CRTDTAARA) again or Rename it (RNMOBJ) back to the data area being checked in your Start Up Program. Only the QSYSWRK subsystem should be started prior to making the independent disk pools available. Then other work will not compete for system resources while your independent disk pools are being made available.

In this example, the data area VARYONIASP is used. You can name your data area whatever you like. Also, in this example the QRECOVERY Library contains the data area; however, you may choose a different library that resides on the system disk pool.

```
MONMSG MSGID(CPF0000)
QSYS/STRSBS SBSDB(QSYSWRK)
QSYS/CHKOBJ OBJ(QRECOVERY/VARYONIASP) OBJTYPE(*DTAARA)
MONMSG MSGID(CPF9801) EXEC(GOTO SKIPVRYCFG)
QSYS/VRYCFG CFGOBJ(IASP1) CFGTYPE(*DEV) STATUS(*ON)
QSYS/VRYCFG CFGOBJ(IASP2) CFGTYPE(*DEV) STATUS(*ON)
SKIPVRYCFG:
```

Switchable and standalone independent disk pools

There are two basic environments in which you can take advantage of independent disk pools: a multi-system environment managed by an iSeries cluster, and a single-system environment with a single iSeries server.

Switchable independent disk pools

Multi-system clustered environment

A group of servers in a cluster can take advantage of the switchover capability within clusters to move access to the independent disk pool from server to server. In this environment, an independent disk pool can be switchable when it resides on a switchable device: an external expansion unit (tower) or an input/output processor (IOP) on the bus shared by logical partitions. The server that owns, or is attached to, the switchable device containing the independent disk pool can then be switched, either automatically in the case of an unplanned outage (failover), or manually by administering a switchover.

Standalone independent disk pools

Single-system environment

An independent disk pool in a single-system environment, with no clustering and no switchable devices, is said to be a private, standalone, or dedicated independent disk pool. While you cannot switch the access to the independent disk pool amongst servers in this environment, you can still isolate data in an independent disk pool, keeping it separate from the rest of the disk storage on the server. The independent disk pool can then be made available (brought online) and made unavailable (taken offline) as you see fit. This might be done, for example, to isolate data associated with a specific application program, or to isolate low-use data that is only needed periodically. This also allows you to isolate certain maintenance functions. Then, when you need to perform disk management functions that normally require the entire system to be at DST, you can perform them by merely varying off the affected independent disk pool.

The following table compares switchable independent disk pools and standalone independent disk pools.

Consideration	Standalone	Switchable	
	Single system	Multi-system cluster	Logical partitions in a cluster
iSeries cluster required	No	Yes	Yes

Connectivity between systems	n/a	HSL loop	Virtual OptiConnect
Location of disk units	Any supported internal or external disk units	External expansion unit (tower)	IOP on shared bus
Switchability	No	Yes - between systems	Yes - between partitions
Switchable entity	None	Expansion unit	IOP

For more on switchable and standalone independent disk pools, including example configurations for each of these environments, see Independent disk pool configurations.

Benefits of independent disk pools

There are two environments in which the use of independent disk pools can be beneficial: a multi-system clustered environment and a single-system environment.

Multi-system clustered environment

In a multi-system clustered environment, where the servers are members of an iSeries cluster and an independent disk pool is associated with a switchable device in that cluster, independent disk pools can be *switched* between systems without having to perform an initial program load (IPL). The independent disk pool can be switched because the independent disk pool is self-contained. This can be a significant advantage as it allows for continuous availability of data, the primary benefit of independent disk pools.

Switchable independent disk pools can help you do the following:

- Keep data available to an application even in the event of a single system outage, either scheduled or unscheduled.
- Eliminate the process of replicating data from one system to another.
- In some situations, isolate disk unit failures within the independent disk pool.
- Achieve high availability and scalability.

Single-system environment

In a single-system environment, where an independent disk pool is privately connected to a single server, independent disk pools can be taken offline, or made unavailable, independent of other disk pools because the data in the independent disk pool is self-contained. The independent disk pool can also be brought online, or made available, while the system is active, without having to perform an IPL. Using independent disk pools this way can be very useful, for example, if you have large amounts of data that are not needed for normal day-to-day business processing. The independent disk pool containing this data can be left offline until it is needed. When large amounts of storage are normally kept offline, you can shorten processing time for operations such as IPL and reclaim storage.

Single-system independent disk pools can help you do the following:

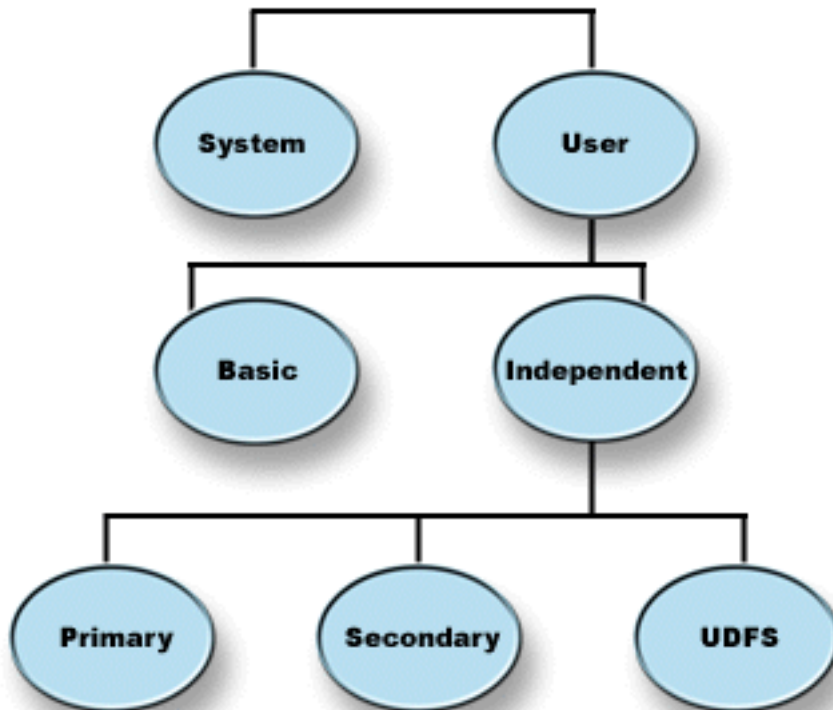
- Isolate low-use data with ability to bring online only when needed.
- Reduce system start time.
- Manage save/restore by independent disk pool.
- Reclaim storage by independent disk pool.
- Divide data between multiple databases.
- Isolate data associated with specific applications or associated with specific groups of users.
- Perform application maintenance that does not affect entire system.

Types of disk pools

An independent disk pool is really a subset of the larger category of disk pools on the iSeries server.

Fundamentally, a disk pool, also referred to as an auxiliary storage pool (ASP), is a software definition of a group of disk units on your system. This means that a disk pool does not necessarily correspond to the physical arrangement of disks. Conceptually, each disk pool on your system is a separate pool of disk units for single-level storage. The system spreads data across the disk units within a disk pool. For more on disk pools (ASPs), see Auxiliary storage pools.

There are two main types of disk pools: system disk pools (system ASPs) and user disk pools (user ASPs). An independent disk pool is a type of user disk pool. The following example and definitions explain the types of disk pools:



System disk pool (system ASP)

One system disk pool exists per iSeries server. The system automatically creates the system disk pool (Disk Pool 1) which contains disk unit 1 and all other configured disks that are not assigned to a basic or independent disk pool. The system disk pool contains all system objects for the OS/400 licensed program and all user objects that are not assigned to a basic or independent disk pool.

User disk pool (user ASP)

There are two types of user disk pools: basic disk pools and independent disk pools. You can create a user disk pool by grouping a set of disk units together and assigning that group to disk pool (ASP). You can configure basic disk pools with numbers 2 through 32. Independent disk pools are numbered 33 through 255. In a clustered environment independent ASPs can be switched between systems without having to perform an IPL, allowing for continuously available data.

Basic disk pool

A basic disk pool is used to isolate some objects from the other objects that are stored in

the system disk pool. Basic disk pools are defined by the user. Data in a basic user pool is always accessible whenever the server is up and running. When storage for a basic disk pool is exhausted, the data can overflow into the system disk pool. This is different from an independent disk pool, which does not allow data to overflow into the system disk pool.

Independent disk pool

A disk pool that contains objects, the directories or libraries that contain the objects, and other object attributes such as authorization and ownership attributes. An independent disk pool can be made available (varied on) and made unavailable (varied off) to the server without restarting the system. When an independent disk pool is associated with a switchable hardware group, it becomes a switchable disk pool and can be switched between one iSeries server and another iSeries server in a clustered environment. An independent disk pool that is not associated with a cluster resource group is referred to in OS/400 application programming interfaces (APIs) as a private disk pool. Independent disk pools can also function in conjunction with other independent disk pools in a disk pool group. The following definitions describe the three types of independent disk pools. There are three types of independent disk pools: user-defined file system, primary, and secondary.

User-defined file system (UDFS)

An independent disk pool that contains only user-defined file systems. It cannot be a member of a disk pool group unless it is converted to a primary or secondary disk pool.

Primary

An independent disk pool that defines a collection of directories and libraries and may have other secondary disk pools associated with it. A primary disk pool also defines a database for itself and other disk pools that may be added in its disk pool group. Primary disk pools can only be implemented on V5R2 or later of OS/400.

Secondary

An independent disk pool that defines a collection of directories and libraries and must be associated with a primary disk pool. A possible use for a secondary disk pool would be to store journal receivers for the objects being journaled in the primary disk pool. Secondary disk pools can only be implemented on V5R2 or later of OS/400.

Disk pool groups

A disk pool group is made up of a primary disk pool and zero or more secondary disk pools. Each disk pool is independent in regard to data storage, but in the disk pool group they combine to act as one entity. If you make one disk pool available or unavailable, the rest of the disk pools in the group are also made available or unavailable at the same time. Also, in a clustered environment, all of the disk pools in a group switch to another node at the same time. The primary and secondary disk pools also share the same database.

An example of a practical use for a disk pool group would be to isolate journal receivers from the objects for which they contain journal entries. The primary disk pool could contain the libraries, journal and objects to be journaled, while the secondary disk pools could contain the associated journal receivers. The journals and journal receivers would remain separate for maximum performance and recoverability, but they would function together in the disk pool group.

Disk pool groups can only be implemented on V5R2 or later of OS/400.

Recommended structure for independent disk pools

The recommended usage structure for independent disk pools is to place the majority of your application data objects into independent disk pools and a minimal number of non-program objects in SYSBAS, which is the system disk pool and all configured basic disk pools. The system disk pool and basic user disk pools (SYSBAS) would contain primarily operating system objects, licensed program product libraries, and very few user libraries. This structure yields the best possible protection and performance. Application data is isolated from unrelated faults and can also be processed independently of other system activity. Vary on and switchover times are optimized with this structure. Other advantages of this structure are:

- No library in the system disk pool is switchable.
- Since a database network cannot span an independent disk pool boundary, entire database networks are contained within disk pool groups.
- Coding of application transactions are simplified since all data libraries are contained within a single disk pool group.
- Library names can be duplicated across disk pool groups, but not between a disk pool group and the libraries in SYSBAS.

Although the above is the recommended structure, this does not exclude other configurations. For example, you may start by migrating only a small portion of your data to a disk pool group and keeping the bulk of your data in SYSBAS. This is certainly supported. However, you should expect longer vary on and switchover times with this configuration since additional processing is required to merge data base cross reference information into the disk pool group.

Warning: Temporary Level 4 Header

Structuring disk pool groups: The iSeries server supports up to 223 independent disk pools, any number of which can be primary, secondary, or UDFS disk pools. Therefore, you have significant flexibility in how you place your data into independent disk pools and how you structure disk pool groups. For example, all application data could be placed in a single disk pool group which consists of one primary disk pool and one secondary disk pool. Alternatively, you could create several disk pool groups, some with only a primary disk pool and some with one or more secondary disk pools.

Consider the following factors when planning the placement of your data in disk pools:

- If an application consists solely of data in user-defined file systems and the data is not to be journaled, a UDFS disk pool might be the best choice. There is less overhead associated with a UDFS disk pool. There is also less extensibility since the UDFS disk pool cannot contain any library-based objects.
- If you have an application with multiple instances of the application data that you want to keep separate, then you should consider a separate disk pool group for each data instance. See Standalone independent disk pools for an example of this scenario.
- If you have multiple applications and the application data is independent, a separate disk pool group for each application might be the appropriate solution. One application's data is then isolated from other applications and each application is unaffected by actions on others. The application data can therefore be brought online, taken offline, or switched without affecting other applications.
- If you have multiple applications with interdependent data objects, the data for those applications should be combined into a single disk pool group.
- You can use secondary disk pools to separate data objects into different storage domains and thus achieve better performance. The normal use of this is to separate your journal receivers onto different disk units from the data being journaled by placing the journal receivers in a secondary disk pool. However, you could also separate other parts of your application onto different disk units providing that they are in different libraries and the following journaling dependency is satisfied.
- Objects being journaled and the journal for those objects must be on the same disk pool.

Independent disk pool restrictions and considerations

Independent disk pools are unique because they are self-contained. All of the necessary system information associated with the data contained on an independent disk pool is contained within. Because of this, there are some restrictions and considerations when using independent disk pools:

- Supported and unsupported OS/400 object types
- Independent disk pools with distinct databases
- Multiple system libraries
- Switching independent disk pool between V5R1 and V5R2 systems
- Object identification
- Printing considerations
- Synchronize user profile name, UID, and GID

Supported and unsupported OS/400 object types

Objects not supported

The following OS/400 objects are NOT supported for use in independent disk pools:

*AUTHLR	*CTLD	*IGCTBL	*NTBD
*AUTL	*DDIR	*IPXD	*NWID
*CFGL	*DEVD	*JOBQ	*NWSD
*CNL	*DOC	*JOBSCD	*OUTQ
*COSD	*EDTD	*LIND	*PRDAVL
*CRG	*EXITRG	*MODD	*USRPRF
*CSPMAP	*FLR	*M36	*SOCKET
*CSPTBL	*IGCSRT	*M36CFG	*SSND
			*S36

Supported object types

The following OS/400 objects are supported for use in independent disk pools:

*ALRTBL	*FIFO	*MODULE	*QRYDFN
*BLKSF	*FILE	*MSGF	*SBSD
*BNDDIR	*FNTRSC	*MSGQ	*SCHIDX
*CHTFMT	*FNTTBL	*NODGRP	*SPADCT
*CHRSF	*FORMDF	*NODL	*SQLPKG
*CLD	*FTR	*OVL	*SQLUDT
*CLS	*GSS	*PAGDFN	*SRVPGM
*CMD	*IGCDCT	*PAGSEG	*STMF
*CRQD	*JOB	*PDG	*SVRSTG
*CSI	*JRN	*PGM	*SYMLNK
*DIR	*JRNRCV	*PNLGRP	*TBL
*DSTMF	*LIB	*PSFCFG	*USRIDX
*DTAARA	*LOCALE	*QMFORM	*USRQ
*DTADCT	*MEDDFN	*QMQR	*USRSPC
*DTAQ	*MENU		*VLDL
*FCT	*MGTCOL		*WSCST

Restrictions for supported object types

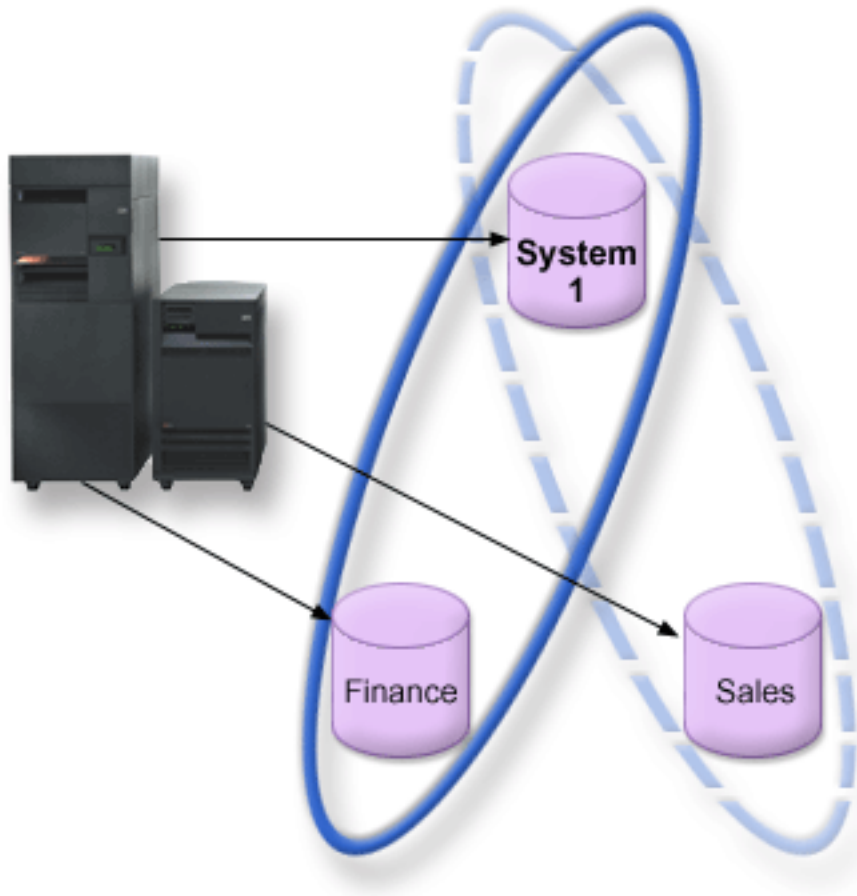
*SBSD

You cannot start a subsystem whose description is located in an independent disk pool.

***FILE** Database files that are either multi-system database files, or that have fields in them that are DataLink fields that are created as Link Control, cannot be located in an independent disk pool.

Independent disk pools with distinct databases

When a primary independent disk pool is configured, a new user database is defined that is separate from the system database. The user database also includes any secondary disk pools that are associated with the primary disk pool. After the primary disk pool is configured, the corresponding user database appears in the Databases folder of iSeries Navigator. By default, the database and the independent disk pool will have the same name. You administer the user database with the same functions that you use for the system database. See *Work with multiple databases* for more information.



The figure above shows an example of a system with three distinct databases: the System database, the independent disk pool Finance database, and the independent disk pool Sales database.



In the example above, if you expand Databases in iSeries Navigator, you see a list of databases that includes the System database as well as the Finance and Sales user databases. From within a user database (Finance and Sales) you can always access libraries in the System database, but you cannot access libraries in another user database. For example, if you open the Finance database, you can select to display libraries from the System database as well. You cannot display Sales libraries from within the Finance database because Sales is a separate user database.

See Object identification for details on identifying objects when independent disk pools exist on your server.

Multiple system libraries

In general, all system libraries will continue to exist in the system disk pool. However, to support better isolation and recovery of the independent disk pool group containing system libraries, the following instances of system libraries will also be created in the primary disk pool:

1. **QSYSnnnnn** — this will contain the database cross reference information for the database represented by the disk pool group. Normally only internal system code creates objects into this library.
2. **QSYS2nnnnn** — this will contain the SQL catalogues for the database represented by the disk pool group. Normally only internal system code creates objects into this library.
3. **QRCYnnnnn** — any recovery object associated with objects within the disk pool group will be stored in this library for the primary disk pool for the group. These objects may be needed for recovery when the disk pool group is varied on. The system disk pool equivalent of this library is QRECOVERY
4. **QRCLnnnnn** — when reclaim is run on the disk pool group, any results information normally stored in QRCL will now be stored in the QRCL of the primary disk pool for the group. Normally only functions called during reclaim storage processing create objects into this library instance. Also, when reclaim storage recovers the addressability of lost objects, these objects can be inserted into the QRCLnnnnn library. These are user objects which originally existed in another library.
5. **QRPLnnnnn** — whenever an object contained within the disk pool group is replaced while it is in use, the in-use object is renamed and moved to the QRPLnnnnn library in the primary disk pool for the group. The new object will be inserted into the specified library. The system disk pool equivalent of this library is QRPLOBJ. QRPLnnnnn is cleared at vary on.

In the above, nnnnn is the independent disk pool number right justified and padded with zeros.

One new library attribute, **Protected**, is introduced to support the extended library capability. Since the libraries QSYSnnnnn, QSYS2nnnnn, and SYSIBnnnnn are special versions that correspond to the system libraries, only operating system code can create objects into them. Applications cannot create objects into these libraries.

The setting of this attribute will be as follows:

Library Attribute Settings

Library	*SYSBAS library	Protected in independent disk pool	Protected in system disk pool
QSYSnnnnn	QSYS	yes	no
QSYS2nnnnn	QSYS2	yes	no
SYSIBnnnnn	SYSIBM	yes	no
QRCLnnnnn	QRCL	no	no
QRCYnnnnn	QRECOVERY	no	no
QRPLnnnnn	QRPLOBJ	no	no
all user libs	n/a	no	no

The normal search order for objects is to search the libraries based on the user specified library value, the user's library list, and the name space in effect for the job. The only exception to this occurs when the user job has a disk pool group in the job's name space. In this case, aliasing support will take affect for object references to the database control objects in QSYS, QSYS2, and SYSIBM. The objects in the QSYSnnnnn, QSYS2nnnnn, and SYSIBnnnnn will actually be returned so that the user is operating on the database control information associated with their extended name space.

Switching independent disk pool between V5R1 and V5R2 systems

Once an independent disk pool is made available on a server that is running OS/400 V5R2, it cannot be made available to a server that is running OS/400 V5R1. It is possible to switch a V5R1 independent disk pool to a V5R2 server and make it available on the V5R2 server. After it is made available on the V5R2 server, its internal contents are changed and it cannot be made available to the V5R1 server again.

Warning: If a V5R2 disk pool is switched to a V5R1 server, its disk units show up as nonconfigured on the V5R1 server. If these disk units are added to another disk pool, the independent disk pool is destroyed.

Object identification

Because the existence of an independent disk pool on a server means that multiple databases will exist on a single server, identifying an object is more complex than it is on a system with only a single system database. When multiple databases exist, it is possible to duplicate the names of libraries and objects in separate databases. The library name and object name don't necessarily uniquely identify an object. There will be times when you'll also need to know the name of the independent disk pool. The name of the independent disk pool and its database are, by default, the same. However, they don't necessarily have to match. A database name can be up to 18 characters long, while an independent disk pool name can be up to 10 characters long.

While the same library name can exist in two different disk pool groups, libraries cannot have the same name in the system disk pool and an independent disk pool.

Control language (CL) commands

When using control language (CL) commands that support specification of *ALL or *ALLUSR for the libraries to be searched, the system will usually interpret this to mean "all (user) libraries in your current library namespace" rather than "all (user) libraries on the system". Some commands may interpret *ALL or *ALLUSR differently so it is important to check the command documentation.

Note: Most messages that go to the job log (QSYSOPR) or history log do not contain the name of the independent disk pool. They only contain the object name and library. You must determine what, if any, disk pool group the job that issued the message was using to be able to find the object.

Printing considerations


If you choose to store external resources for spooled files, not the spooled files themselves, in a disk pool group, you must be aware of the printing implications. You can store formatting objects such as *FNTRSC, *FORMDF, *OVL, *PAGDFN, and *PAGSEG in a disk pool group. In order for the printer writer job to access these objects, you must set the disk pool group so that it exists in the library name space.

Follow these steps to print the spooled file when external resources are stored in a disk pool group:

1. Make sure that the disk pool group that contains the external resources is available.
2. Set the disk pool group for the current thread using the SETASPGRP (Set ASP Group) Command (disk-pool-group-name).
3. Print the spooled file using the STRPRTWTR (Start Printer Writer) Command (printer-device-name).

Plan for independent disk pools

There are several requirements that must be satisfied in order to implement independent disk pools, particularly if you plan to use switchable independent disk pools. Setting up an environment for switching devices begins with careful planning.

Important: When you are ready to order a new server or a server upgrade to implement clusters, IBM will assist you in making sure that your cluster requirements are met. See Planning for Clustering. 

Creating a standalone, or dedicated, independent disk pool does not require as much planning as a switchable independent disk pool. However, you should still take the time to make sure that your future needs will not require you to be able to switch the independent disk pool.

See the following for details on the requirements for independent disk pools:

- Hardware requirements
- Physical planning requirements
- Software and licensing requirements
- Communications requirements
- Cluster requirements

Hardware requirements

Depending upon how you plan to implement independent disk pools, you must have the following hardware:

Multi-system clustered environment (for switchable independent disk pools)

1. Two or more iSeries servers capable of running OS/400 V5R1M0¹ or later.
-or-
One iSeries server capable of running OS/400 V5R1M0¹ or later configured with logical partitions (LPAR)
2. One or more switchable devices. This could be:
 - One or more expansion units (towers) residing on an HSL loop
 - One or more input/output processors (IOP) in a logical partition

Note: In an LPAR environment, you can switch the input/output processor (IOP) containing the independent disk pools between system partitions without having an expansion unit. The IOP must be on the bus shared by multiple partitions. All input/output adapters (IOAs) on the IOP will be switched.
-

Single system environment

One iSeries server capable of running OS/400 V5R1M0 or later.

¹ OS/400 V5R1M0 can be used for implementing independent disk pools containing user-defined file systems (UDFS) only. Support for library-based objects is only available starting with OS/400 V5R2M0.

Physical planning requirements

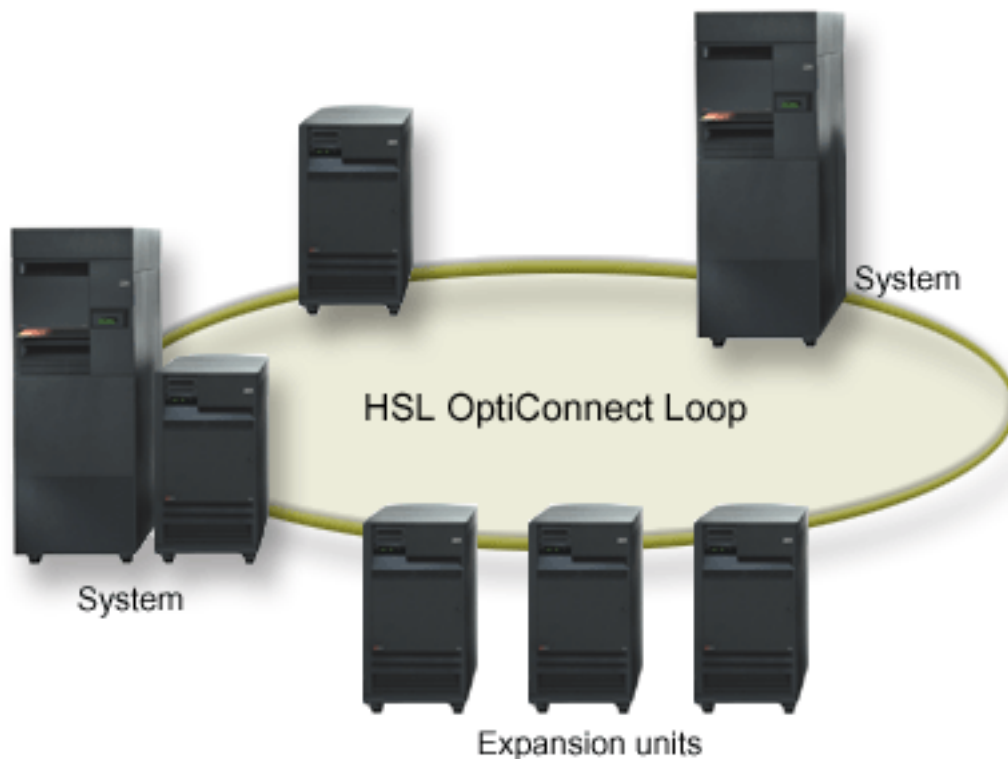
Depending upon how you plan to implement independent disk pools, you must satisfy the following physical planning requirements:

Multi-system clustered environment (for switchable independent disk pools)

High Speed Link (HSL) cables must be used to attach the expansion units (towers) to the servers in the cluster.

The expansion unit must be physically adjacent in the HSL loop to the alternate system or expansion unit owned by the alternate system. You can include a maximum of two servers (cluster nodes) on each HSL loop, though each server can be connected to multiple HSL loops. You can include a maximum of four expansion units on each HSL loop, though a maximum of three expansion units can be included on each loop segment. On an HSL loop containing two servers, two segments exist, separated by the two servers. All expansions on one loop segment must be contained in the same device CRG.

The switchable expansion unit must be SPCN-cabled to the system unit that will serve as the primary node for the switchable hardware group (device CRG). The primary node could be a primary or secondary logical partition within the system unit. If using LPAR, the system buses in the intended tower must be owned dedicated by the partition involved in the cluster.



Single system environment

There are no physical planning requirements.

Software and licensing requirements

Depending upon how you plan to implement independent disk pools, you must have the following software and licenses:

Multi-system clustered environment (for switchable independent disk pools)

1. OS/400 V5R1M0¹ or later
2. iSeries Navigator
iSeries Navigator is the graphical user interface for managing and administering your iSeries server from your Windows(R) desktop. It is required to perform some of the disk management tasks necessary to implement independent disk pools. See Access disk management functions for steps to enable iSeries Navigator for disk management.
3. Option 41 (OS/400 - HA Switchable Resources) - installed and licensed
Option 41 is a clustering requirement that gives you the capability to switch independent disk pools between systems. In order to switch an independent disk pool between servers, the servers must be members of a cluster and the independent disk pool must be associated with a switchable hardware group in that cluster. Option 41 also gives you the capability of using the IBM Simple Cluster Management interface in iSeries Navigator to define and manage a simple cluster that uses switchable resources.

Single system environment

1. OS/400 V5R1M0¹ or later
2. iSeries Navigator
iSeries Navigator is the graphical user interface for managing and administering your iSeries server from your Windows(R) desktop. It is required to perform some of the disk management tasks necessary to implementing independent disk pools. See iSeries Navigator and independent disk pools for details.

¹ OS/400 V5R1M0 can be used for implementing independent disk pools containing user-defined file systems (UDFS) only. Support for library-based objects is only available starting with OS/400 V5R2M0.

Communications requirements

Depending upon how you plan to implement independent disk pools, you must satisfy the following communications requirements:

Multi-system clustered environment (for switchable independent disk pools)

Switchable independent disk pools are configured within an iSeries Cluster. The communication requirement for a cluster environment is at least one TCP/IP communication interface between the servers in the cluster. For redundancy, it is recommended that there be two separate interfaces between the servers.

NOTE: It is not required that the HSL OptiConnect Loop interface between servers be used in a switchable expansion unit (tower) configuration. Also, it is not required that Virtual OptiConnect communication between LPAR partitions be used in a switchable IOP in a logical partition environment.

Single system environment

There are no communications requirements.

Cluster requirements

If you plan to implement *switchable* independent disk pools, you will need to configure an iSeries cluster. The documentation in these independent disk pools topics will guide you through the creation and management of your cluster. However, you may want to prepare your network and server environment in advance.

Use the Cluster configuration checklist to ensure that you are prepared to configure clusters in your environment.

Application considerations for independent disk pools

When you are designing or restructuring your application environment for use with independent disk pools, there are several things you should be aware of. A few of these considerations include: the existence of multiple databases, the objects that can and cannot be created in an independent disk pool, how the library list works, and the placement of programs and data in the correct database.

When a primary independent disk pool is made available for the first time, a new database with the same name is also generated by default. See *Independent disk pools with distinct databases* for more information. If you write an application to access files and libraries in a disk pool group, you must specify how to access that specific database. Some of your options include:

- Use the Set ASP Group (SETASPGRP) command.
- In an SQL environment use CONNECT to specify the right database. To achieve the fastest performance, make sure that the database to which you perform an SQL CONNECT corresponds with your current library name space. You may need to use the SETASPGRP command first to achieve this. If the SQL CONNECT function is not operating within the same library name space, the application will use Distributed Relational Database Architecture^(TM) support which can affect performance.
- Use the Change Job Description (CHGJOB) command to set the initial ASP group in the job description for a user profile.

As you write applications that create objects, you must know which objects are supported and unsupported. If your application uses the Create Library (CRTLIB) command, you must specify CRTLIB ASP(*ASPDEV) ASPDEV(*asp-device-name*). If you do not specify these parameters for CRTLIB, the library is created in the system disk pool by default. However, if you use the SQL statement, CREATE COLLECTION, the default for the IN ASP clause is the current library name space.

Another thing to be aware of when you are operating in an SQL environment is that permanent SQL objects cannot span independent disk pool boundaries. For example, you cannot create a view of an independent disk pool object in the system disk pool. This action fails.

A similar concept is true for commitment control with independent disk pools. If you are connected to an independent disk pool relational database, you cannot make committable changes against objects in any other disk pool. When commitment control is active, you have read only access. You can make committable changes against QTEMP, but you may receive error messages.

It may also be helpful to understand how the library list works when independent disk pools are implemented. When the library list includes QSYS, QSYS2, or SYSIBM, the aliased libraries in the independent disk pool (QSYSnnnnn, QSYS2nnnnn, SYSIBnnnnn) are searched before the libraries in the system disk pool. If the object is found in the independent disk pool, the system disk pool will not be searched. In addition, if you switch to a different disk pool group, any libraries that were in the previous library list are removed from the current library list.

You also need to carefully consider where you store data, applications, and application exit programs. It is recommended that data should be stored in independent disk pools. If your independent disk pools are dedicated to the server, it may work to store applications and exit programs in the system database so that they are always accessible, no matter what disk pool group is associated with a job. If you use the independent disk pool in a clustered environment, you must remember that when the disk pool is switched to another server, the exit program must be available there as well. In this case, it may be more appropriate to store the applications and exit programs in the independent disk pool. Remember that the cluster resource group (CRG) exit program cannot exist in an independent disk pool.

If you are using the independent disk pool in a clustered environment, you must also remember that the user profiles are not stored in the independent disk pool. They are kept in the system disk pool. If an independent disk pool fails over or is switched to another node, a user profile may be created on the new node. For a user profile to be created, it must own objects or be authorized to objects in the primary disk pool of the disk pool group that is being switched. The new user profile will not have special authorities or a password.

If you are operating in a clustered environment, see Cluster applications for more information about writing and implementing highly-available applications within your cluster.

Configure independent disk pools

Once you have satisfied the planning requirements for implementing independent disk pools, you are ready to configure an independent disk pool. You will need to use the iSeries Navigator disk management function to configure an independent disk pool. See Access disk management functions for details.

See the following topics to get your independent disk pools configured:

- Access disk management functions
Complete the steps to access the required disk management functions in iSeries Navigator.
- Create a switchable independent disk pool
Independent disk pools can be switchable between servers in an iSeries cluster.
- Create a standalone independent disk pool
See this topic for creating an independent disk pool that will be privately connected to a single system.
- Create a disk pool group
A disk pool group is made up of a primary disk pool and zero or more secondary disk pools. A practical use of a disk pool group would be to isolate journal receivers, which would reside in one or more secondary disk pools, from the objects for which they contain journal entries, which would reside in the primary disk pool.
- Convert UDFS disk pools
If you have existing user-defined file system (UDFS) disk pools on your server, you can convert them to primary and secondary disk pools, allowing them to support library-based objects.

Access disk management functions

iSeries Navigator is the graphical user interface for managing and administering your iSeries server from your Windows^(R) desktop. You can use iSeries Navigator wizards and dialogs to create and manage your independent disk pool environment. See iSeries Navigator for information on the capabilities, requirements, and installation of iSeries Navigator.

Before you can access disk management functions in iSeries Navigator, you must complete the following steps:

Install the Configuration and Service component

1. From the **File** menu of iSeries Navigator select **Install Options** —> **Selective Setup**.
2. Follow the instructions on the resulting dialog to install the Configuration and Service component.

Enable the Disk Units folder

1. In iSeries Navigator right-click the server connection and select **Application Administration**.
2. On the resulting window, click **OK**.
3. Click the **Host Applications** tab.
4. Expand **Operating System/400** —> **Service**.
5. Select **Disk Units** to have **Default Access** or **All Object Access**.
6. Click **OK**.

7. Restart iSeries Navigator.

Configure the service tools server

To access disk management functions in iSeries Navigator, you must first configure the service tools server with DST access and user IDs. Be familiar with Service tool concepts before you start. See [Configure the service tools server](#) and [Configure service tools user IDs](#) for instructions.

Disk management

Disk management functions are available in the Disk Units folder of iSeries Navigator. Follow these steps to access disk management functions in iSeries Navigator:

1. In iSeries Navigator, expand **My Connections**.
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.

For more planning tips such as steps to access iSeries Navigator disk management functions in Dedicated Service Tools (DST) mode, use the graphical view, and calculate disk space, see [Planning for disk management](#).

Create a switchable independent disk pool


Before you attempt to implement switchable independent disk pools, ensure that you have satisfied the hardware, software, communications, and physical planning requirements. See [Plan for independent disk pools](#).

iSeries Navigator is the recommended interface for creating and managing independent disk pools. Wizards in the clusters and disk management components simplify the tasks and guide you through the process. For some disk management tasks, iSeries Navigator is the only option. Make sure you can [Access disk management functions in iSeries Navigator](#). If you use control language (CL) commands and application program interfaces (APIs), there are additional steps which are handled internally when using iSeries Navigator.

Using iSeries Navigator

1. **Create a cluster.**
To use switchable independent disk pools, an iSeries cluster is required.
2. **Make hardware switchable.**
If you have a standalone tower or an IOP that contains disk units that are to be included in an independent disk pool, you must authorize the tower or IOP to grant access to other nodes.
3. **Create a switchable hardware group.**
A switchable hardware group, also known as a device CRG, defines the switchable independent disk pool. This is what manages the switching of the device. This wizard takes you through the steps to create a new switchable hardware group. It will also guide you through the New Disk Pool wizard which will assist you in creating a new disk pool and adding disk units to it for the cluster.
Note: If you had switchable software products which conform to specific iSeries Navigator cluster guidelines installed when you ran the New Cluster wizard in step 1, the New Cluster wizard may have already prompted you to create a switchable hardware group. If the New Cluster wizard did not detect that a switchable software product was installed, then you have not created the switchable hardware group.

4. **Print your disk configuration.**

Print your disk configuration to have in case of a recovery situation. See How to display your disk configuration in Backup and Recovery.  Also, record the relationship between the independent disk pool name and number.

* You have now created a switchable independent disk pool. The remaining steps are required to prepare it for use.

5. **Make the disk pool available.**

To access the disk units in an independent disk pool you must make the disk pool available (vary on) the disk pool.

6. **Start the switchable hardware group.**

Start the switchable hardware group to enable device resiliency for the switchable hardware group.

7. **Perform a test switchover.**

Before you add data to the disk pool, perform a test switchover on the switchable hardware group you created to ensure the configuration functions as you planned.

Using CL commands and APIs

You can use CL commands and APIs for creating a switchable independent disk pool, however there are some tasks that require that you use iSeries Navigator.

1. **Create the cluster.**

Create the cluster with desired nodes using the CRTCLU (Create Cluster) Command.

2. **Create the device domain.**

You must create the device domain for all nodes involved in switching an independent disk pool or set of independent disk pools using the ADDDEVDMNE (Add Device Domain Entry) command.

3. **Create the device descriptions.**

Device descriptions must be created on each node that will be in the cluster resource group (CRG). Use the CRTDEVASP (Create Device Description (ASP)) Command. On the command line in the character-based interface, enter CRTDEVASP. In the **Resource Name** and the **Device Description** fields, enter the name of the independent disk pool you plan to create.

4. **Create the cluster resource group.**

Create the device CRG with the nodes, their roles in the recovery domain, and independent disk pool device descriptions using the CRTCRG (Create Cluster Resource Group) Command.


5. **Make hardware switchable.**

If you have a standalone tower or an IOP that contains disk units that are to be included in an independent disk pool, you must authorize the tower or IOP to grant access to other nodes (**iSeries Navigator required**).

6. **Create the switchable independent disk pool.**

Create the disk pool on the node that owns the disk units using the New Disk Pool wizard when the server is fully restarted. Make sure clustering is active before you start. Name the independent disk pool to match the device description resource name that you gave in step 3. As you add disk units, it is best to localize disk units in the same tower or IOP. Also, do not spread the disk pool across device parity sets (**iSeries Navigator required**).

7. **Print your disk configuration.**

Print your disk configuration to have in case of a recovery situation. See How to display your disk configuration in Backup and Recovery.  Also, record the relationship between the independent disk pool name and number.

* You have now created a switchable independent disk pool. The remaining steps are required to prepare it for use.

8. **Make the disk pool available.**
To access the disk units in an independent disk pool you must make the disk pool available (vary on) the disk pool (**iSeries Navigator required**).
9. **Start the cluster resource group.**
Start the cluster resource group to enable device resiliency using the STRCRG (Start Cluster Resource Group) Command.
10. **Perform a test switchover.**
Before you add data to the disk pool, perform a test switchover to ensure the configuration functions as you planned.

You are now ready to populate the independent disk pool with directories and libraries. Before you do, be sure to read Independent disk pools with distinct databases.

Create a cluster

In order for an independent disk pool to be switchable amongst servers, an iSeries cluster is required. An iSeries cluster is a collection or group of one or more servers that work together as a single server. For complete documentation on clusters and how they work, see Clusters.

There are several solutions available for creating and managing a cluster. You can use iSeries Navigator to create a simple cluster, a cluster middleware business partner solution, or IBM cluster commands and APIs. See Solutions for configuring clusters for complete look at the options for configuring and managing clusters.

To create a cluster for use with switchable independent disk pools:

1. **Create a cluster.**
For step-by-step instructions on how to create a cluster, see Create a cluster in the Clusters topic.
2. **Verify that all nodes are at potential cluster version 3 and the current cluster version must be set to 3.**
See Adjust the cluster version of a cluster for details.
3. **Start all nodes in the cluster, or at least those that will be in the device domains.**
See Start a cluster node for details.

Make your hardware switchable

An independent disk pool can contain disk units within several expansion units (towers). If you have a standalone tower that contains disk units included in an independent disk pool, you must authorize the tower to grant access to other servers. This is called making a tower switchable. If you do not want other servers to be able to access the standalone tower, you must make the tower private.

Make a tower switchable

To make a tower switchable, follow these steps:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. Expand **By Location** and select the towers you want to make switchable.
7. Right-click a highlighted tower and select **Make Switchable**.

8. Follow the instructions on the dialog that displays.

Change a bus ownership type

To allow an IOP to be switched, the bus containing the IOP that controls the disk units to be switched must be *owned shared* by the primary node. The bus must also be *use bus shared* by the backup node. See Dynamically switching IOPs between partitions for more information.

To complete this task, you need a Service Tools user profile with administration authority to the System Partitions function in Dedicated Service Tools (DST). For more information on obtaining logical partition privileges, refer to Logical partition authority.

To change the ownership type for a bus using Management Central, follow these steps:

1. In iSeries Navigator, expand **My Connections**.
2. Select the primary partition of the system.
3. Expand **Configuration and Service** and select **Logical Partitions**.
4. Right-click the **Logical Partition** and select **Configure Partitions**. You are now working in the Configure Logical Partitions window.
5. Right-click the bus for which you want to change ownership and select **Properties**.
6. Select the **Partitions** page.
7. Select the partition that owns the bus in **Owning logical partition**, and then select the ownership type in **Sharing**. If the ownership type is shared, the partitions that share the bus appear in the list. Click Help if you need more information on these options.
8. Click OK.

Create a switchable hardware group

A switchable hardware group, also known as a device cluster resource group (CRG), contains a list of switchable devices. Each device in the list identifies a switchable independent disk pool. The entire collection of devices are switched to the backup node when an outage, planned or unplanned, occurs. Optionally, the devices can also be made available (varied on) as part of the switchover or failover process.

A switchable hardware group identifies a device domain. A device domain is simply a subset of cluster nodes that share a set of resilient devices. The device domain is created automatically when you use the iSeries Navigator wizard to create a cluster. If you are using cluster CL commands and APIs, you must add each node that you want to be switchable to the device domain.

Using iSeries Navigator

requires Option 41 (OS/400 - HA Switchable Resources)

The New Switchable Hardware Group wizard will take you through the steps to create a new switchable hardware group and add a disk pool to it for the cluster.

To add a switchable hardware group, follow these steps:

1. In iSeries Navigator, expand **Management Central**.
2. Expand **Clusters**.
3. Expand the cluster for which you would like to add a switchable hardware group.
4. Right-click **Switchable Hardware**, and select **New Group...**
5. By default the New Disk Pool wizard creates a protected disk pool that allows you to choose how you want to protect the disk units. You can use device parity protection, mirrored protection, or a combination of both. After the disk pool is created, you will be prompted to start mirroring. This ensures that if you make changes to your disk pool configuration, it will remain protected. You can also also create an unprotected disk pool by unchecking the protection option.

Note: Make sure that all the nodes in the recovery domain are started.

Using Cluster CL commands and APIs

You can also use the following to add a device domain entry and create a device cluster resource group:

Add Device Domain Entry

Adds a node to a device domain membership list so that it can participate in recovery actions for resilient devices. The addition of the first node to a device domain has the effect of creating that device domain.

- ADDDEVDMNE (Add Device Domain Entry) Command
- Add Device Domain Entry (QcstAddDeviceDomainEntry) API

Create Cluster Resource Group

Creates a cluster resource group object. The cluster resource group object identifies a recovery domain, which is a set of nodes in the cluster that will play a role in recovery.

- CRTCRG (Create Cluster Resource Group) Command
- Create Cluster Resource Group (QcstCreateClusterResourceGroup) API

Make a disk pool available

To access the disk units in an independent disk pool and the objects in the corresponding database you must make the disk pool available (vary on).

In a multi-system clustered environment, you can make the disk pool available to the current node or to another node in the cluster. The independent disk pool can only be varied on to one node at a time. When you want to access the independent disk pool from a different node, you must switch the independent disk pool to the backup cluster node. See Perform a switchover for details on switching a device CRG (referred to as a switchable hardware group in iSeries Navigator) to the backup node.

Note: If you make a primary or secondary disk pool available, all of the disk pools in the disk pool group will also be made available at the same time.

To make an independent disk pool available:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. Sign on to service tools if the Service Tools Signon dialog displays.
7. Expand **Disk Pools**.
8. Right-click the unavailable disk pool and select **Make Available**. You can select multiple disk pools to make available at the same time.
9. From the dialog displayed, click **Make Available** to make the disk pool available.

You can also use the Vary Configuration (VRYCFG) command in the character-based interface to make the disk pool available.

Start switchable hardware group

To enable device resiliency for the switchable hardware group, you must start the switchable hardware group.

To start a switchable hardware group, follow these steps:

1. In iSeries Navigator, expand **Management Central**.
2. Expand **Clusters**.
3. Expand the cluster that contains the switchable hardware you would like to start.
4. Click **Switchable Hardware**.
5. Right-click the switchable hardware group you would like to start, and select **Start**.

You can also use the Start Cluster Resource Group (STRCRG) command in the character-based interface to start the switchable hardware group.


Create a standalone independent disk pool

Creating a standalone, or dedicated, independent disk pool does not require as much planning and configuration as a switchable independent disk pool requires. However, you should still take the time to make sure that your future needs will not require you to be able to switch the independent disk pool.

To create a standalone independent disk pool, you can use the New Disk Pool wizard in iSeries Navigator. This will assist you in creating a new disk pool and adding disk units to it. The New Disk Pool wizard also allows you to include nonconfigured disk units in a device parity set, and start device parity protection and disk compression. As you add disk units, do not spread disk units that are in different parity sets across multiple disk pools. Make sure you can Access disk management functions in iSeries Navigator.

To use the New Disk Pool wizard to create a standalone independent disk pool, follow these steps:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. Right-click **Disk Pools** and select **New Disk Pool**.
7. Follow the wizard's instructions to add disk units to a new disk pool.
8. When you have completed the New Disk Pool wizard, print your disk configuration to have in case of a

recovery situation. See [How to display your disk configuration in Backup and Recovery](#).  Also, record the relationship between the independent disk pool name and number.

Note: Add independent disk pools when your server is fully restarted. If you must use the New Disk Pool wizard at the dedicated service tools (DST) level, you need to create an associated device description for the independent disk pool when the server is fully restarted. Use the Create Device Description (ASP) (CRTDEVASP) command to create the device description; name the device description and resource name the same as you name the independent disk pool. You can use the Work with Device Descriptions (WRKDEVD) command to verify that the device description and independent disk pool name match.

Create a new disk pool group

You can create a disk pool group and add disk units to the individual disk pools by using the New Disk Pool wizard. If you have existing UDFS disk pools that you would like to include in a disk pool group, see [Convert a UDFS disk pool to primary](#) or [Convert a UDFS disk pool to secondary](#).

Note: If you want to create a switchable independent disk pool (UDFS, Primary, or Secondary), you must create the cluster first. For more information, see [Create a switchable independent disk pool](#).

To create a new disk pool group, follow these steps:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. Right-click **Disk Pools** and select **New Disk Pool**.
7. On the resulting New Disk Pool dialog, select **Primary** for the Type of Disk Pool field and complete the required information.

Note: If you have already created a primary disk pool with which you want to associate one or more secondary disk pools in a disk pool group, you can skip this step. After you have created the primary disk pool, click New Disk Pool if you want to create a secondary disk pool to associate with the primary disk pool. From the resulting dialog, select Secondary for the Type of Disk Pool field and complete the required information. Repeat this step for each secondary disk pool you want to create. Follow the wizard's instructions to add disk units to the new disk pools.

Convert UDFS disk pools

Support for library-based objects through the use of primary and secondary disk pools was introduced at V5R2. If you have existing user-defined file system (UDFS) disk pools on your server, you can convert them to primary and secondary disk pools. This will allow them to support library-based objects.

You must convert UDFS disk pools if you want them to participate in a disk pool group. Once you convert a UDFS disk pool to a primary or secondary disk pool, you cannot convert it back to a UDFS disk pool. You must create a primary disk pool before you can associate secondary disk pools.

To accomplish this conversion, see:

- Convert a UDFS disk pool to primary
- Convert a UDFS disk pool to secondary

Manage independent disk pools

Once you have configured an independent disk pool, you can perform management tasks using iSeries Navigator. Make sure you can Access disk management functions.

Some of the tasks that you may need to perform include:

- Backup and recovery
Be sure to consider a save strategy for your independent disk pools.
- Delete an independent disk pool
You can select an independent disk pool to delete.
- Make a disk pool available
To access the disk units in an independent disk pool you must make the disk pool available (vary on).
- Make a disk pool unavailable
You can select an independent disk pool to make unavailable (vary off).
- Make your hardware switchable
In a multi-system environment, you must make an external expansion unit (tower) switchable.
- Recover an independent disk pool
If problems occur in a disk pool, you can attempt to recover it.
- Switch access to backup server
Perform a cluster switchover when you want a backup server to access the switchable device containing an independent disk pool

- Change the server takeover IP address
Change the IP address for a server associated with a relational database in a clustered, switchable environment.
- Synchronize user profile name, UID, and GID
Synchronize user profiles across your cluster to reduce the amount of processing required when you make a disk pool available.

Make a disk pool unavailable

You can select an independent disk pool to make unavailable (vary off). You will not be able to access any of the disk units or objects in the independent disk pool or its corresponding database until it is made available (varied on) again. The pool can be made available again on the same system or another system in the recovery domain of the cluster resource group.

Important: Before an independent disk pool can be made unavailable, no jobs can hold reservations on the disk pool. See Release job reservations on an independent disk pool for details on determining whether jobs are using an independent disk pool and how to release the job reservations.

To make an independent disk pool unavailable:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. Sign on to service tools if the Service Tools Signon dialog displays.
7. Expand **Disk Pools**.
8. Right-click the disk pool you want to make unavailable and select **Make Unavailable**.
9. From the dialog that displays, click **Make Unavailable** to make the disk pool unavailable.

You can also use the Vary Configuration (VRYCFG) command in the character-based interface to make the disk pool unavailable.

Recover an independent disk pool

If you are experiencing problems accessing an independent disk pool or making it available, there may be a problem with the disk pool. Possible problems include:

- The configuration source is corrupted. When corruption occurs, the independent disk pool will appear to have no disk units in it. The disk pool may also appear to have no disk units in it if it is switched to another node in a clustered environment. Before you attempt a recovery, make sure that no other system owns the disk pool. If you know the serial numbers of the disk units in the independent disk pool that may need recovery, make sure you are on the system that owns those disk units and that they show as non-configured.

If the configuration source is corrupted, you can select to recover the configuration information on the configuration source. Recovering the configuration attempts to determine the original configuration and recover it. During this process, the independent disk pool may need to be cleared, destroying all data on the disk units in the pool. If the disk pool needs to be cleared, a message displays warning you of this and allowing you to cancel the recovery.

- The mirrored disk unit of the configuration source is damaged. When this happens, the mirrored configuration source becomes unknown. The disk pool will be unavailable, and you must recover the configuration information of an unknown configuration source before making it available. You should only attempt to recover the state of the unknown configuration source when you know its mirrored disk unit was active before the failures that caused the state to become unknown.

To attempt to recover an independent disk pool, follow these steps:

1. In iSeries Navigator, expand **My Connections** (or your active environment).
2. Expand any iSeries server.
3. Expand **Configuration and Service**.
4. Expand **Hardware**.
5. Expand **Disk Units**.
6. If the Service Tools Signon dialog displays, sign on to service tools.
7. Select **Disk Pools**.
8. Right-click the problematic disk pool. If iSeries Navigator detects one of the problems listed above, then **Recover Configuration** or **Recover Unknown Configuration Source** appears in the list. If you see either of these options, select it to continue.
9. Follow the instructions on the dialog displayed.

Switch access to backup server

In a multi-system clustered environment that uses switchable independent disk pools, an independent disk pool can only be accessed by one node at a time. Current access to a switchable independent disk pool is managed through the switchover function within the cluster.

To switch access from the current node in the cluster to the backup node:

1. Make the disk pool unavailable (vary off) from the current node. (This step is optional. The switchover processing in the next step attempts to make the disk pool unavailable if it is currently available.)
2. Switch the independent disk pool to the backup cluster node by performing a switchover in the cluster. See Perform a switchover for details on switching a device CRG (referred to as a switchable hardware group in iSeries Navigator) to the backup cluster node.

Change the server takeover IP address

The server takeover IP address is associated with a primary disk pool in a clustered, switchable environment. Specifically, it is the IP address for a server associated with the relational database name in the device description for a switchable independent disk pool. The specified address must exist on all nodes in the recovery domain if the cluster resource group is active.


To change the server takeover IP address for a primary disk pool, follow these steps:

1. In iSeries Navigator, expand **Management Central**.
2. Expand **Clusters**.
3. Expand the cluster that contains the switchable hardware group.
4. Expand **Switchable Hardware**.
5. Click the switchable hardware group, then right-click the desired primary disk pool and select **Properties**.
Note: The server takeover IP address can only be associated with a primary switchable independent disk pool.
6. Change the server takeover IP address in the **IP address** field.

You can also use the CHGCRGDEVE (Change Cluster Resource Group Device Entry) command in the character-based interface to change the server takeover IP address.

Backup and recovery of independent disk pools

A good save strategy is just as important for independent disk pools as it is with the rest of your system information. If you use independent disk pools, it is recommended that you use Backup, Recovery and Media Services (BRMS) to save your independent disk pool data. If you need to perform a recovery, BRMS simplifies the process. However, BRMS is not required; see Save independent ASPs for more

information. In the case of disk failures or a complete system loss, you may need to follow recovery procedures to restore the data you have saved. See the Backup and Recovery  manual for steps to restore information to the independent disk pools.

If you experience problems accessing an independent disk pool or making it available, there may be a problem with the disk pool. It may be that the configuration source is corrupted or that the primary and secondary disk pools needs to be re-associated. See the following topics for steps to recover the disk pools:

- Recover an independent disk pool
- Recover a disk pool group

Examples: Independent disk pools configurations

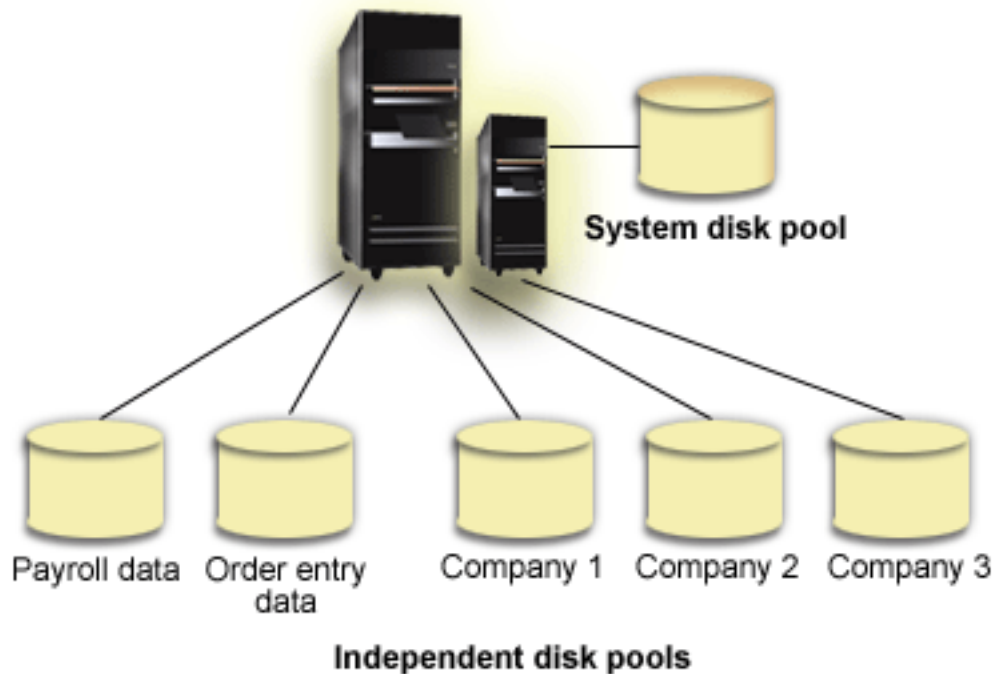
Independent disk pools can be switchable amongst a group of servers in a cluster, providing the benefits of continuous availability of the disk units they contain. Or they can be standalone, or dedicated, on a single server, independent of the rest of the storage on the server.

See the following for examples of each type of independent disk pool implementations:

- Standalone independent disk pools
- Switchable independent disk pools

Standalone independent disk pools

In a single-system environment, a standalone, or dedicated, independent disk pool can be taken offline independent of other disk pools because the data in the independent disk pool is self-contained. That is, all of the necessary system information associated with the independent disk pool's data is contained within the independent disk pool. The independent disk pool can also be brought online while the system is active; no initial program load (IPL) required. Using independent disk pools this way can be very useful, for example, if you have large amounts of data that are not needed for normal day-to-day business processing. The independent disk pool containing this data can be left offline until it is needed. When large amounts of storage are normally kept offline, you can shorten processing time for operations such as IPL and reclaim storage.



In this scenario, the user has five independent disk pools. They could represent three different applications where the third application may have archived data. The system automatically creates the system disk pool (referred to as *Disk Pool 1* or *ASP 1*) which contains all system programs and system data.

Switchable independent disk pools

In a multi-system environment, an independent disk pool can be switched between servers in a cluster. A switchable independent disk pool is a set of disk units that you can switch between servers so that each server can access the data. Only one system can access the data at a time.

Switchable independent disk pools can reside on one of two types of switchable hardware devices:

External tower (expansion unit)

The switchable device can be an external tower (expansion unit) connected to the clustered servers on the same High-Speed link (HSL) loop.

Input/output processor (IOP) in a logical partition

In an LPAR environment, the switchable device can be an IOP on the bus shared by the partitions.

The entity that switches is actually the tower or the IOP containing the independent disk pool. When a tower or IOP is switched, all of the hardware attached to the switchable entity is moved to the backup system.

The following example configurations illustrate some typical switchable independent disk pools implementations:

Example: Switchable tower

This example features an implementation with four servers and two switchable towers. In a collection of single iSeries servers, with no logical partitions configured, you can switch a tower containing the independent disk pools between adjacent systems. The tower and the systems must be on the same HSL loop.

Example: Switchable IOP with logical partitions

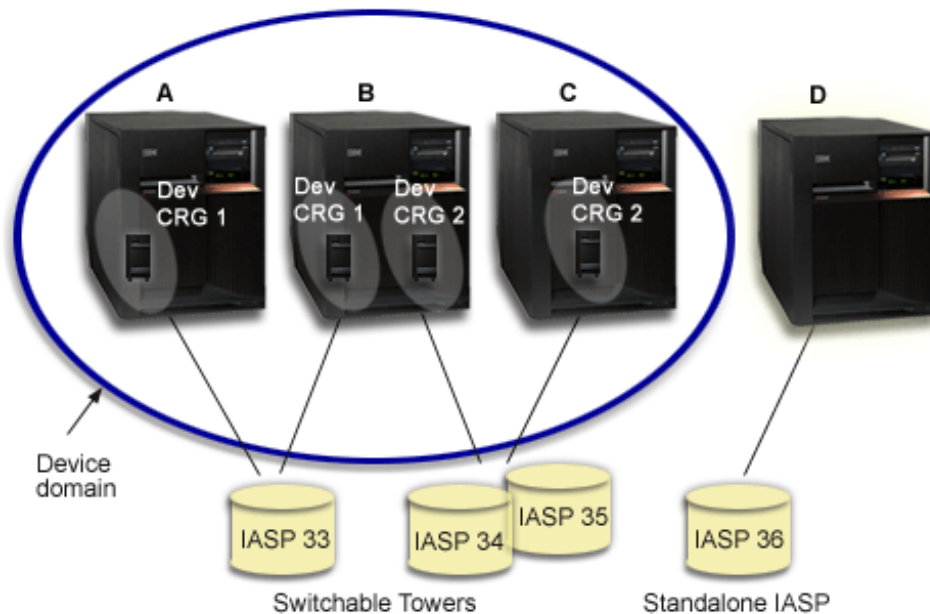
This implementation consists of of four logical partitions and a switchable IOP. When an iSeries is configured with logical partitions, you can switch the IOP containing the independent disk pools between system partitions. The IOP can be on the bus shared by the partitions or it could be in an external tower shared by multiple processors.

Example: Switchable tower with logical partitions

You can also use a combination of the previous examples by switching a tower between logical partitions. This example depicts a combination of both a switchable tower and iSeries servers configured with logical partitions. Only the tower is switchable. No switchable IOP is present.

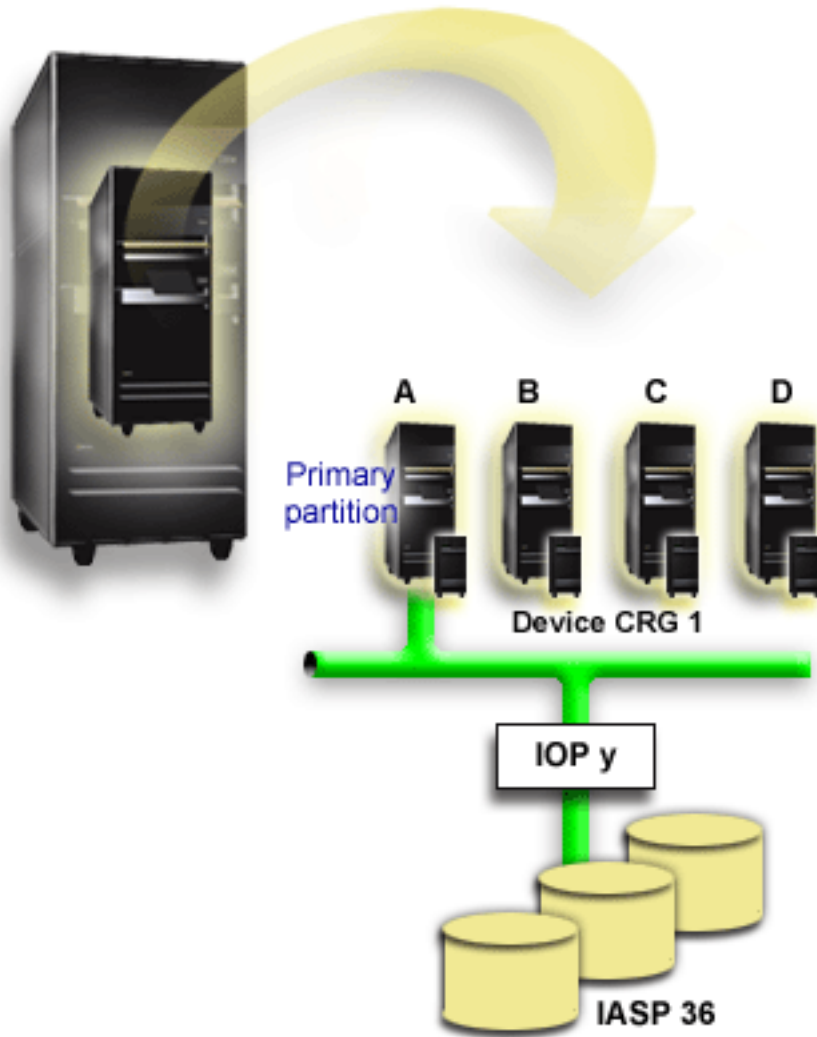
Example: Switchable tower

In this example, the following figure shows a cluster consisting of four nodes. Nodes named A, B, and C are defined to be in the same device domain. There are two switchable towers - one contains IASP33 and the other contains IASP34 and IASP35. The tower containing IASP33 is on an HSL loop that also contains nodes A and B. This first tower can be switched between nodes A and B. The tower containing IASP34 and IASP35 could be on another HSL loop that also contains nodes B and C. This second tower can be switched between nodes B and C. Node D is contained in the cluster, but is not a member of the device domain and therefore can only access IASP36, a standalone, or dedicated, independent disk pool.



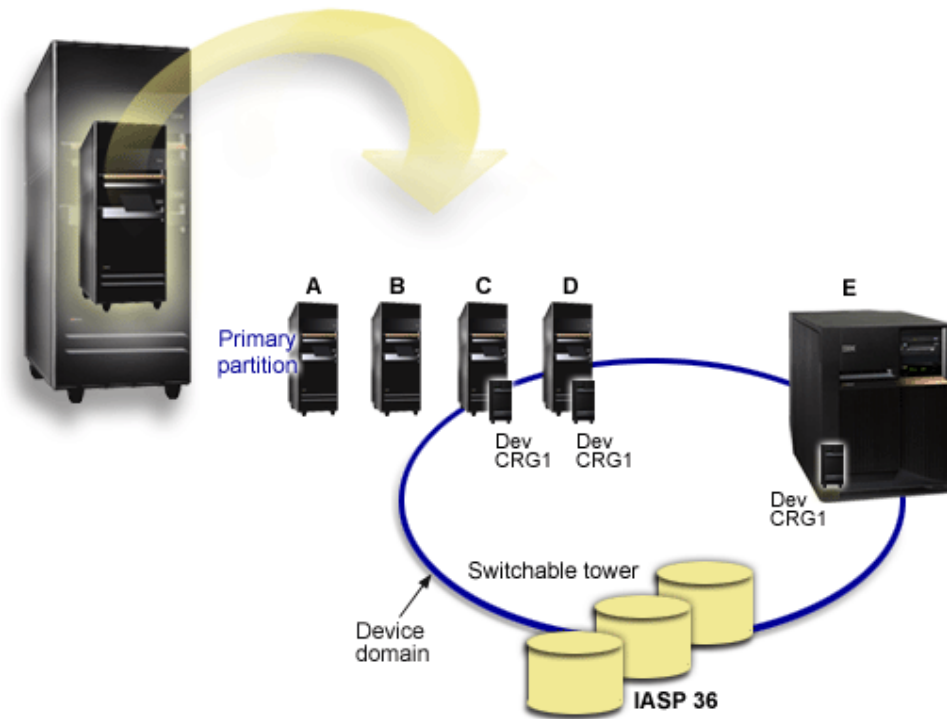
Example: Switchable IOP with logical partitions

In this logical partition example, the following figure shows a cluster consisting of four logical partitions on a single iSeries server. All four nodes belong to the same device domain. IASP36 is composed of disk units accessible through IOP Y. IOP Y is on the shared bus so it can be switched between all of the nodes in the cluster: A, B, C, and D. When the IOP is switched, everything that is physically connected to that IOP is also moved to the new primary node.



Example: Switchable tower with logical partitions

The example, shown in the figure below, depicts a combination of the previous two examples. IASP36 is composed of disk units contained in a switchable tower. The tower is on the same HSL loop as two systems, one of which is made up of four logical partitions. Assuming that nodes C and D, and the second server, node E, are defined to be in the same device domain, the independent disk pool can be switched between those three nodes.



Frequently asked questions

Here is a list of independent disk pool questions and answers. If you have a question that is not on this page, please contact us.

General

1. 34
2. How can independent disk pools be implemented in my environment? (See 34)
3. How should I structure my independent disk pools? (See 34)
4. What is a disk pool group? (See 34)

iSeries Navigator graphical user interface

1. How do I access the iSeries Navigator disk management function? (See 34)
2. What's the difference between the disk management functions in iSeries Navigator and the character-based command interface? (See 34)
3. How do I access the disk management function when the system is at the dedicated service tools (DST) level? (See 35)
4. What is the Service Tools Server (STS)? (See 35)
5. Why does the data I see in iSeries Navigator appear to be out of date? (See 35)
6. Why can't I connect to the service tools server after I added the service table entry? (See 35)

Configuring

1. How do I create a new disk pool or independent disk pool? (See 35)
2. How do I create a disk pool group? (See 36)

Performance

1. Why is performance slow? (See 36)

Troubleshooting

1. Why do no disk units appear as eligible to be added to my disk pool? (See 36)
2. Why doesn't the device description get deleted when I delete the disk pool? (See 36)
3. Why do I get a warning message saying the device description is already created? (See 36)
4. Why does the primary or secondary disk pool I tried to create appear to be a UDFS disk pool? (See 36)
5. Why do I get a message that says my disk pool is not the right type when I try to create a library in it? (See 36)

Warning: Temporary Level 3 Header

General

How do independent disk pools work?

The key characteristic of an independent disk pool is its ability to be, of course, independent of the rest of the storage on a server. It is independent because the data in the independent disk pool is self-contained. This means that all of the necessary system information associated with the data resides within the independent disk pool. See *How independent disk pools work* for details.

[Back to questions \(See 33\)](#)

How can independent disk pools be implemented in my environment?

There are two basic environments in which you can take advantage of independent disk pools: a multi-system environment managed by an iSeries cluster, and a single-system environment with a single iSeries server. See *Switchable and standalone independent disk pools* for details.

[Back to questions \(See 33\)](#)

How should I structure my independent disk pools?

IBM provides some recommendations for structuring and populating your independent disk pools. See *Recommended structure for independent disk pools* for details.

[Back to questions \(See 33\)](#)

What is a disk pool group?

A disk pool group is made up of a primary disk pool and zero or more secondary disk pools. Each disk pool is independent in regard to data storage, but in the disk pool group they combine to act as one entity. See *Disk pool groups* for details.

[Back to questions \(See 33\)](#)

iSeries Navigator graphical user interface

How do I access the iSeries Navigator disk management function?

Before you can access disk management functions in iSeries Navigator, you must complete some setup tasks. See *Access disk management functions* for details.

[Back to questions \(See 33\)](#)

What's the difference between the disk management functions in iSeries Navigator and the character-based command interface?

Support for many independent disk pools tasks are only available through iSeries Navigator. Almost all disk management functions that are available from the system service tools (SST) level are available through iSeries Navigator. A number of disk management functions that are only available from the dedicated service tools (DST) level are also available.

[Back to questions \(See 33\)](#)

How do I access the disk management function when the system is at the dedicated service tools (DST) level?

Beginning with V5R1, the Disk Units container in iSeries Navigator is available when the system is at the dedicated service tools (DST) level.

[Back to questions \(See 33\)](#)

What is the service tools server (STS)?

The service tools server allows you to use your PC to perform service tools functions through TCP/IP. Before you attempt to use any disk management functions, you must configure the service tools server. See Set up communication for disk management for details.

[Back to questions \(See 33\)](#)

Why does the data I see in the iSeries Navigator window appear to be out of date?

The disk management function in iSeries Navigator caches information, and therefore needs to be refreshed to have the most current data visible. After the you make a configuration change, iSeries Navigator should refresh itself. If it does not, however, you may manually refresh it by clicking the Refresh button on the iSeries Navigator toolbar. You can also set iSeries Navigator to refresh periodically. Depending upon the size of your server, however, you may not want to do this. Disk unit configuration data tends to be fairly static, and so does not need to be refreshed often. If your system is very large, it can take a significant amount of time to download all information.

[Back to questions \(See 33\)](#)

Why can't I connect to the service tools server after I added the service table entry?

The Add Service Table Entry (ADDSRVTBLE) command is case sensitive. In particular, it is important to ensure that the Protocol = 'tcp', and not 'TCP'. To ensure this is the case, use the Work with Service Table Entry (WRKSRVTBLE) command, and check the as-sts server field. Make sure that TCP is lower case. If it is not, then remove then entry, and recreate it by issuing the following command exactly:

```
ADDSRVTBLE SERVICE('as-sts') PORT(3000) PROTOCOL('tcp') TEXT('Service Tools Server')  
ALIAS('AS-STs')
```

[Back to questions \(See 33\)](#)

Configuring

How do I create a new independent disk pool?

You can create an independent disk pool in a clustered, multi-system environment or on a single system. See the following topics for details:

- Create a switchable independent disk pool
- Create a standalone independent disk pool

[Back to questions \(See 33\)](#)

How do I create a disk pool group?

See Create a new disk pool group for details.

[Back to questions \(See 33\)](#)

Performance

Why is performance slow?

There are several factors that can influence performance. Make sure your PC's TCP/IP settings are configured correctly. Specifically, make sure that you do not have an invalid secondary gateway. If you do have a secondary gateway, remove it. This should provide a significant increase in performance.

[Back to questions \(See 33\)](#)

Troubleshooting

Why do no disk units appear as eligible to be added to my disk pool?

There are a number of possible reasons for this. First, you must have a non-configured disk unit to add. If the disk pool is protected, you will only be able to add parity disks, or disks in pairs, so they can be mirrored.

If your system is in a clustered environment, things get a bit more tricky. Each disk unit is assigned a **Rank** which indicates its eligibility to be added to a particular disk pool. If the Rank of the disk unit is above 300, then the disk is ineligible. A complete list of the ranks, and what they mean, is available in the disk management online help.

[Back to questions \(See 33\)](#)

Why doesn't the device description get deleted when I delete the disk pool?

Because the device description doesn't always get created by disk management function, it may not be deleted when the disk pool gets deleted. You will have to manually delete it using the Delete Device Description (DLTDEVD) command.

[Back to questions \(See 33\)](#)

Why do I get a warning message saying the device description is already created?

When you create a new independent disk pool, an attempt is made to create an associated device description. If a device description of the same name as the disk pool already exists, you will see a warning message, and the existing device description will not be modified. Most of the time, this is not a problem. However, if the device description's name and associated resource do not match, this becomes a problem, and this is why you see the warning message.

[Back to questions \(See 33\)](#)

Why does the primary or secondary disk pool I tried to create appear to be a UDFS disk pool?

If iSeries Navigator crashed or was closed while the disk pool was being created, you may need to Convert the UDFS pool to a primary or secondary.

[Back to questions \(See 33\)](#)

Why do I get a message that says my disk pool is not the right type when I try to create a library in it?


Make sure that the disk pool you are trying to create a library in is a primary or secondary disk pool, not a UDFS disk pool. If the disk pool is a UDFS disk pool and you want to create a library in it, you need to Convert the UDFS pool to a primary or secondary disk pool.


[Back to questions \(See 33\)](#)

Related information

Listed below are the web sites and IBM Redbooks™ that relate to independent disk pools:

Web sites

High Availability and Clusters 
IBM site for High Availability and Clusters

Learning Services US 
IBM site for IT product training, custom solutions, and e-Learning. You can search for courses offered on clustering and independent disk pools.

Redbook

Clustering and IASPs for Higher Availability  (about 6.4 MB or 330 pages)
This redbook presents an overview of cluster and switched disk technology available for iSeries servers.

iSeries IASPs - A guide to working with Independent Auxiliary Storage Pools 
This redpiece presents a step-by-step approach to independent ASPs on iSeries servers.



Printed in U.S.A.