

# ບົດທີ 9

ການປັບປະສິດທິພາບຂອງດິດໃຫ້ເໝາະສົມ  
(Disk Performance Optimization)

# ເນື້ອໃນຫຍໍ້

- ◆ ສະເໜີເບື້ອງຕົ້ນ
- ◆ ວິວັດທະນາການຂອງໜ່ວຍຄວາມຈໍາສໍາຮອງ
- ◆ ລັກສະນະການເຄື່ອນທີ່ຂອງຫົວອ່ານ
- ◆ ຄວາມຈໍາເປັນໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk
- ◆ ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk
- ◆ ການປັບການໝູນ Disk
- ◆ ສິ່ງທີ່ຄວນພິຈາລະນາ
- ◆ Caching and Buffering
- ◆ ການປັບປະສິດທິພາບຂອງ Disk ແບບອື່ນໆ

# ສະເໜີເບື້ອງຕົ້ນ

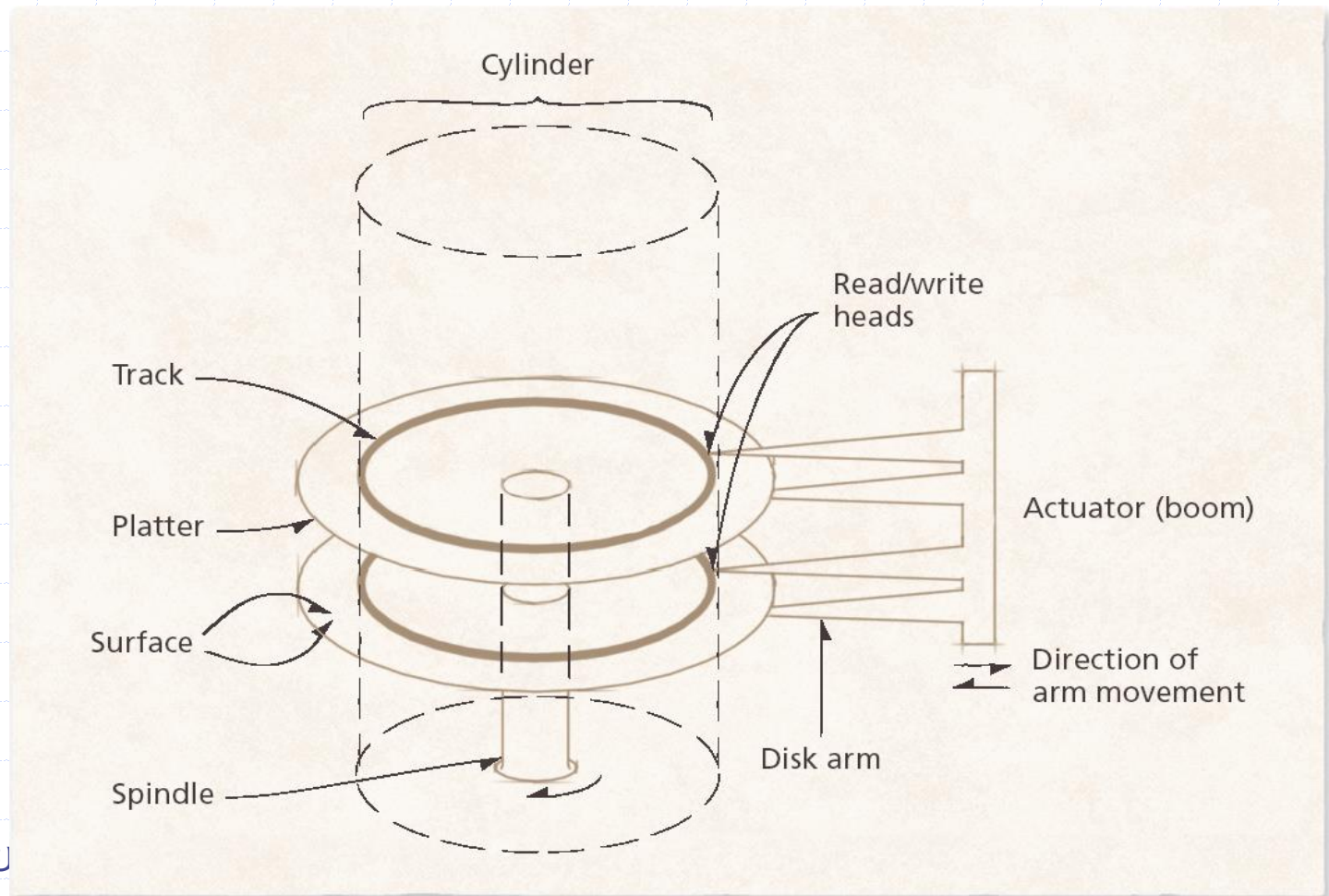
- ◆ ໜ່ວຍຄວາມຈຳສຳຮອງເປັນຕົ້ນເຫດອັນໜຶ່ງທີ່ເຮັດໃຫ້ລະບົບຊຳລົງໄປເລື້ອຍໆ
- ◆ ການເຮັດໃຫ້ໜ່ວຍຄວາມຈຳສຳຮອງເຮັດວຽກໄວຂຶ້ນຈະຊ່ວຍໃຫ້ປະສິດທິພາບໂດຍລວມຂອງລະບົບໄວຂຶ້ນ
- ◆ ສາມາດແກ້ໄຂບັນຫາດັ່ງກ່າວໂດຍໃຊ້ ຊອບແວຣ໌ ແລະ ຮາດແວຣ໌

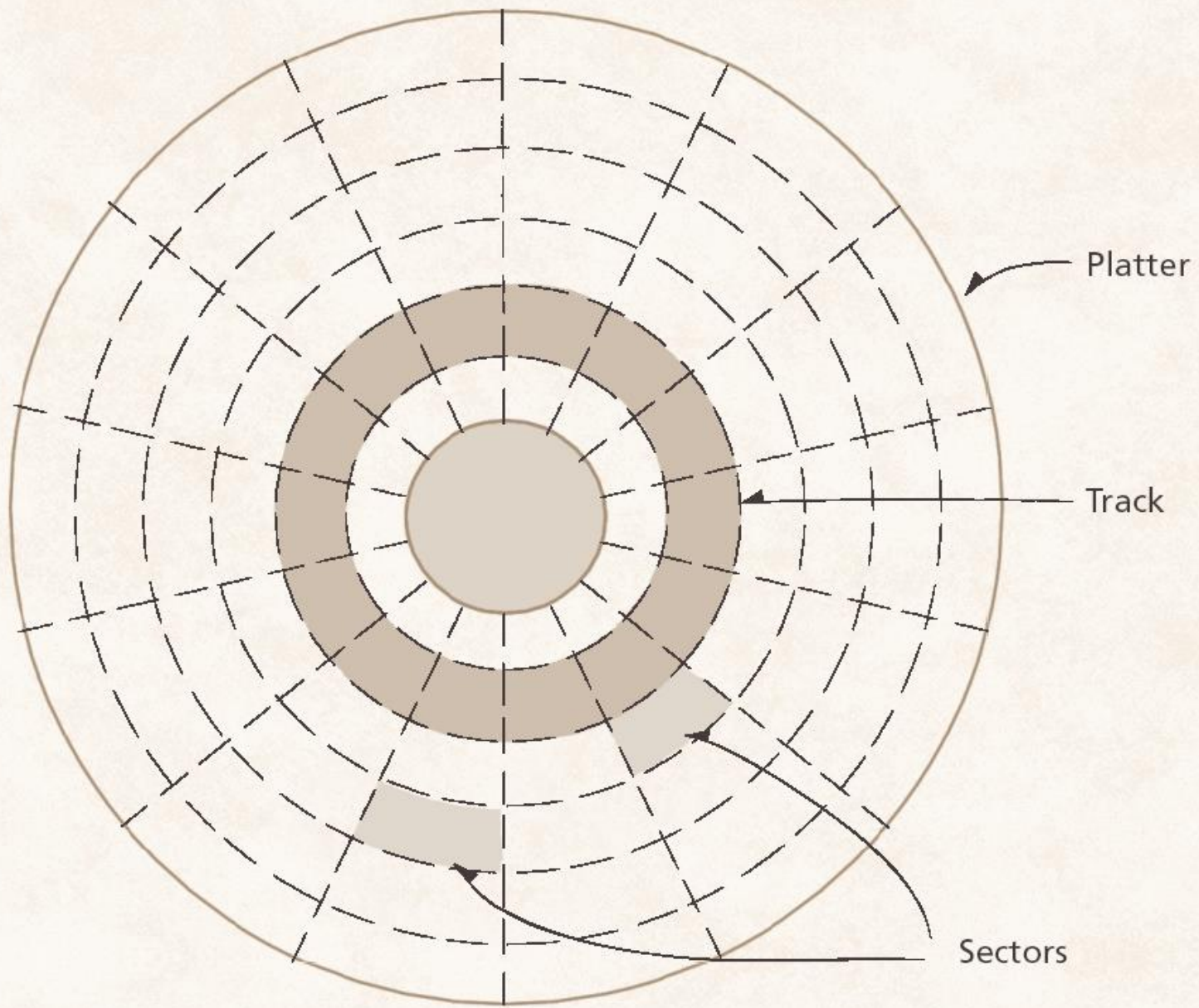
# ວິວັດທະນາການຂອງໜ່ວຍຄວາມຈໍາສໍາຮອງ

- ◆ ໜ່ວຍຄວາມຈໍາສໍາຮອງເກືອບທັງໝົດແມ່ນເກັບຂໍ້ມູນໄວ້ໃນລັກສະນະທີ່ແມ່ເຫຼັກ
- ◆ ການເຂົ້າຫາຂໍ້ມູນແມ່ນອາໄສຫົວອ່ານ-ຫົວຂຽນ
- ◆ ເທັກໂນໂລຢີສະໃໝກ່ອນ, ການເຂົ້າຫາຂໍ້ມູນເປັນແບບຕາມລໍາດັບ (Sequential Access)
  - ຊຶ່ງເປັນການເຂົ້າຫາຂໍ້ມູນອັນຕໍ່ອັນໄປຕາມລໍາດັບ
  - ເຮັດໃຫ້ບໍ່ໄດ້ປະສິດທິພາບສໍາຫລັບ applications ແບບເຂົ້າຫາໂດຍກົງ
- ◆ ເທັກໂນໂລຢີໃນຍຸກໃໝ່ຈະເປັນແບບ Random-access
  - ສາມາດເອີ້ນອີກຢ່າງໜຶ່ງວ່າ direct-access
  - ຊຶ່ງສາມາດເຂົ້າຫາຂໍ້ມູນແຕ່ລະຕົວແບບໂດຍກົງໄດ້

# ລັກສະນະການເຄື່ອນທີ່ຂອງຫົວອ່ານ

## ◆ ຮູບຮ່າງທາງກາຍຍະພາບ





# ລັກສະນະການເຄື່ອນທີ່ຂອງຫົວອ່ານ

## ◆ ການວັດແທກປະສິດທິພາບ

### ■ Seek time

- ◆ ເປັນເວລາທີ່ໃຊ້ໃນການເຄື່ອນຍ້າຍຫົວອ່ານ-ຫົວຂຽນໄປຫາ track/Cylinder ທີ່ມີຂໍ້ມູນທີ່ຕ້ອງການ

### ■ Rotational latency

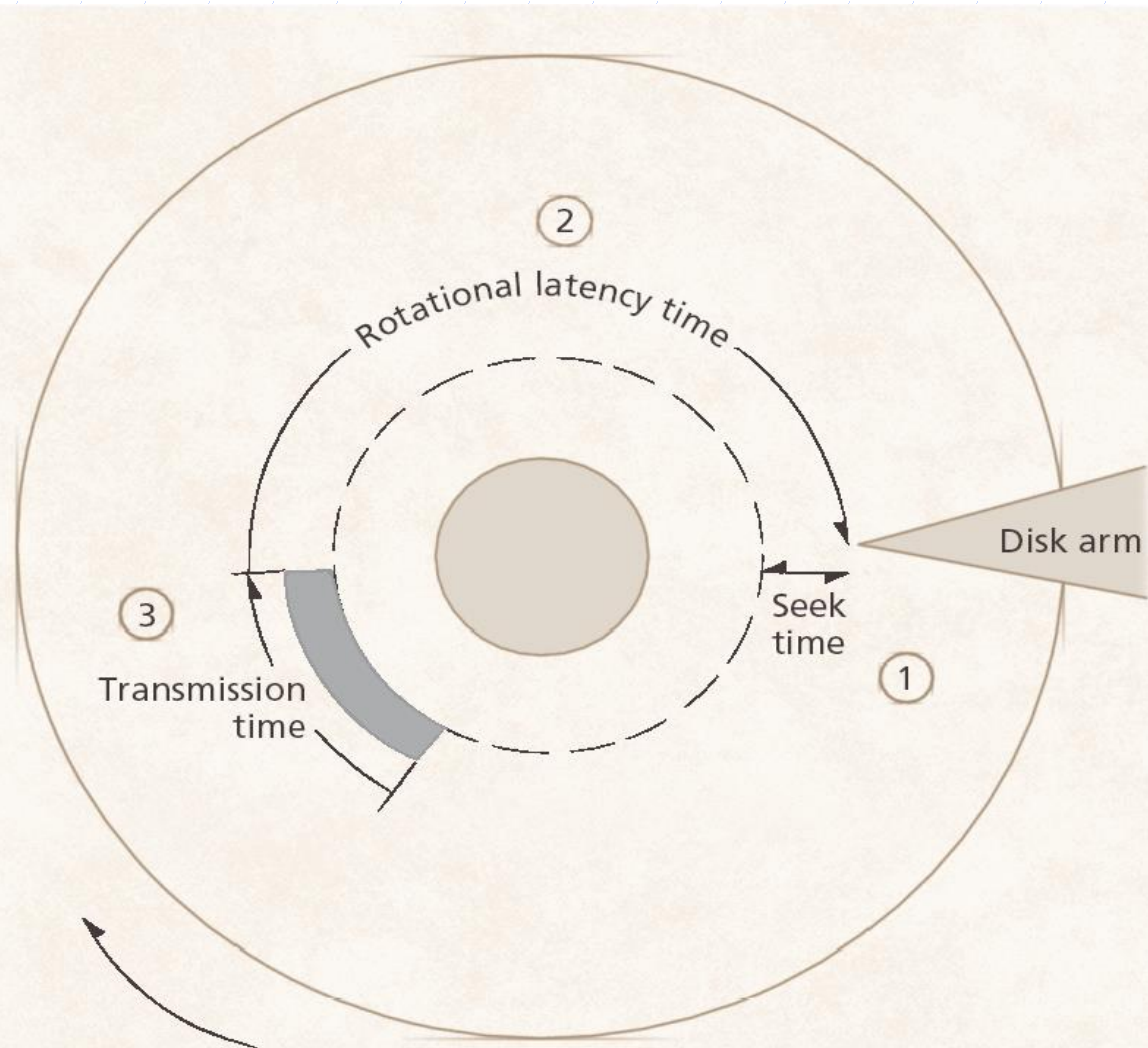
- ◆ ເປັນເວລາທີ່ໃຊ້ໃນການໝູນເອົາຂໍ້ມູນມາຫາຫົວອ່ານ-ຫົວຂຽນ

### ■ Transmission time

- ◆ ເປັນເວລາທີ່ໃຊ້ໃນການອ່ານເອົາຂໍ້ມູນດັ່ງກ່າວແລ້ວສົ່ງມາຫາໜ່ວຍປະມວນຜົນ



# ລັກສະນະການເຄື່ອນທີ່ຂອງຫົວອ່ານ





# ຄວາມຈຳເປັນໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

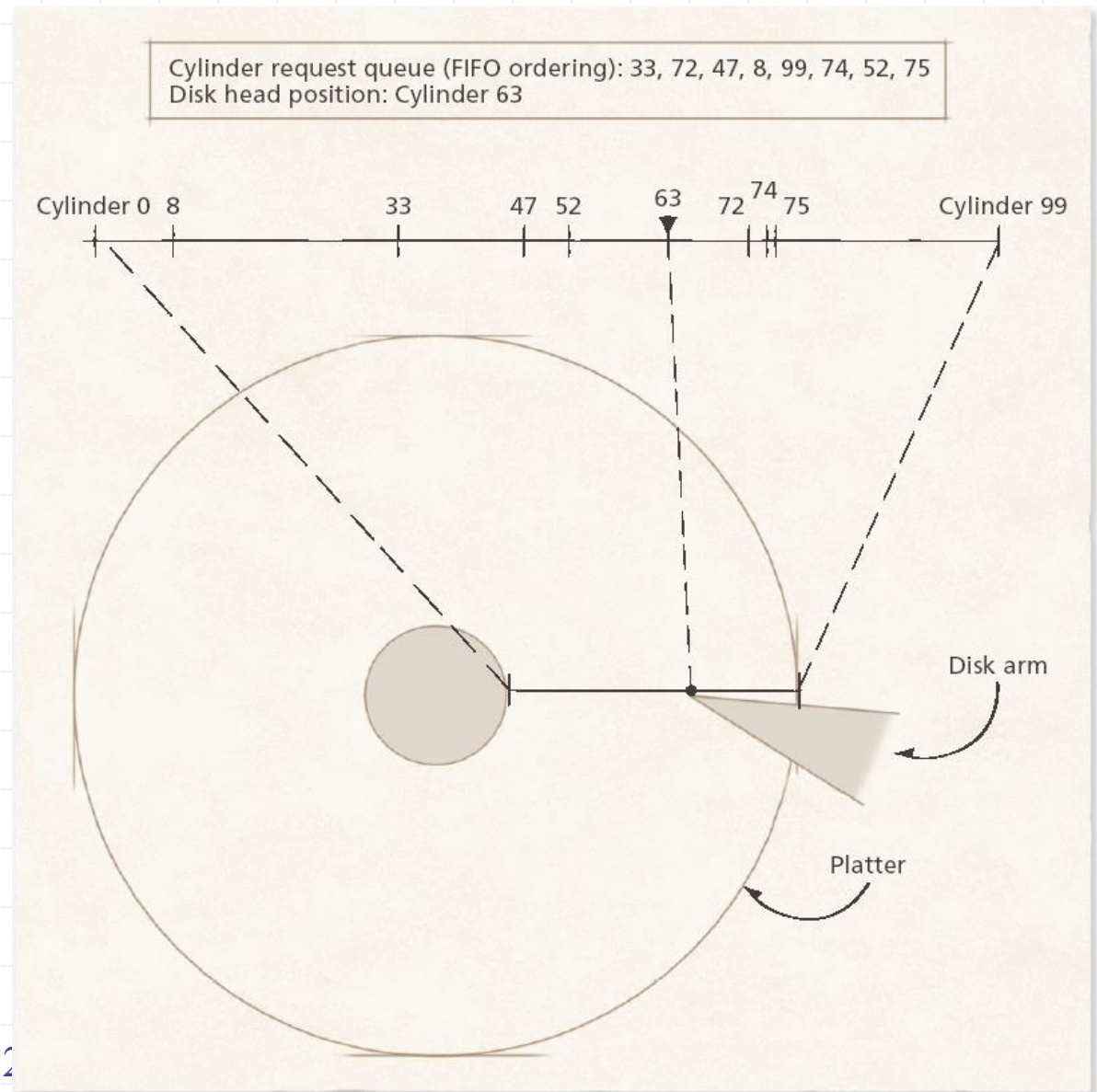
- ◆ ແມ່ນການໄປກວດສອບເບິ່ງແຕ່ລະການຂໍຮ້ອງເພື່ອກຳນົດວິທີໃນການໃຫ້ບໍລິການໃຫ້ມີປະສິດທິພາບທີ່ສຸດ
- ◆ ເຮັດໃຫ້ການອ່ານ, ການຂຽນຂໍ້ມູນເຮັດໄດ້ໃນປະລິມານທີ່ຫຼາຍຂຶ້ນໃນຫົວໜ່ວຍເວລາໃດໜຶ່ງ
- ◆ ຕົວຈັດການຕາຕະລາງຮາດດິດຈະເປັນຜູ້ກວດສອບເວລາໃນການລໍຖ້າຂອງແຕ່ລະຂະບວນການ ແລ້ວໄປຈັດລຳດັບຄືນໃໝ່ເຮັດແນວໃດໃຫ້ມີການເຄື່ອນຍ້າຍຫົວອ່ານໃນໄລຍະທີ່ໄກທີ່ສຸດ
- ◆ ໃນການເຮັດໃຫ້ຫຼຸດເວລາລໍຖ້າແມ່ນຈະຕ້ອງຫຼຸດເວລາໃນການເຄື່ອນຍ້າຍຫົວອ່ານ ແລະ ຫຼຸດເວລາໃນການປິ່ນເອົາຂໍ້ມູນມາຫາຫົວອ່ານ

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

- ◆ ມີ 3 ປັດໃຈທີ່ໃຊ້ວັດແທກຍຸດທະສາດ
  - ຈຳນວນການໃຫ້ບໍລິການໃນຫົວໜ່ວຍເວລາໃດໜຶ່ງ (Throughput)
  - ເວລາໃນການລໍຖ້າໂດຍສະເລ່ຍ (Mean response time)
  - ເວລາທີ່ມີການປ່ຽນແປງ (Variance of response times)
- ◆ ເປົ້າໝາຍ
  - ໃຫ້ສາມາດໃຫ້ບໍລິການໄດ້ຫຼາຍທີ່ສຸດໃນຫົວໜ່ວຍເວລາໃດໜຶ່ງ
  - ເຮັດເວລາລໍຖ້າໂດຍສະເລ່ຍ ແລະ ເວລາທີ່ປ່ຽນແປງນ້ອຍທີ່ສຸດ

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

ຮູບແບບໃນການຂໍ  
ຮອງຂໍ້ມູນຈາກດິດ



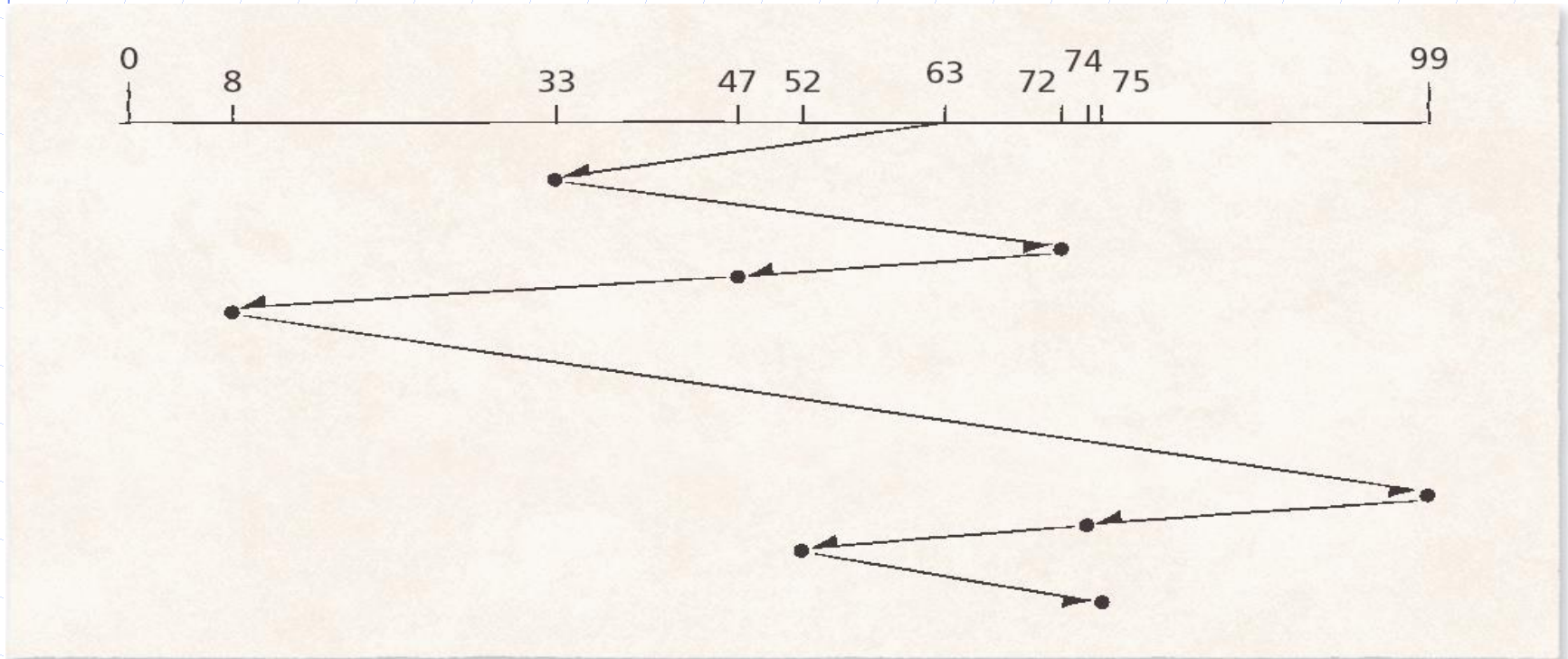
# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການຈັດຕາຕະລາງແບບມາກ່ອນ-ໄດ້ກ່ອນ (FCFS Scheduling)

- ການຂໍຮ້ອງໄດ້ໃຫ້ບໍລິການຕາມລຳດັບທີ່ຂໍຮ້ອງມາ
- ຈຸດດີ
  - ◆ ຍຸດຕິທຳ
  - ◆ ບໍ່ເຮັດໃຫ້ເກີດການເລື່ອນການເຮັດວຽກໄປຕະຫຼອດການ
  - ◆ ພັດທະນາງ່າຍ
- ຜົນເສຍ
  - ◆ ອາດຈະເຮັດໃຫ້ໄດ້ບໍລິການໜ້ອຍທີ່ສຸດໃນຫົວໜ່ວຍເວລາໃດໜຶ່ງ

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການຈັດຕາຕະລາງແບບມາກ່ອນ-ໄດ້ກ່ອນ (FCFS Scheduling)



Queue: 33, 72, 47, 8, 99, 74, 52, 75. Head position: 63

ຮູບແບບການໃຫ້ບໍລິການດ້ວຍວິທີການມາກ່ອນ-ໄດ້ກ່ອນ

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

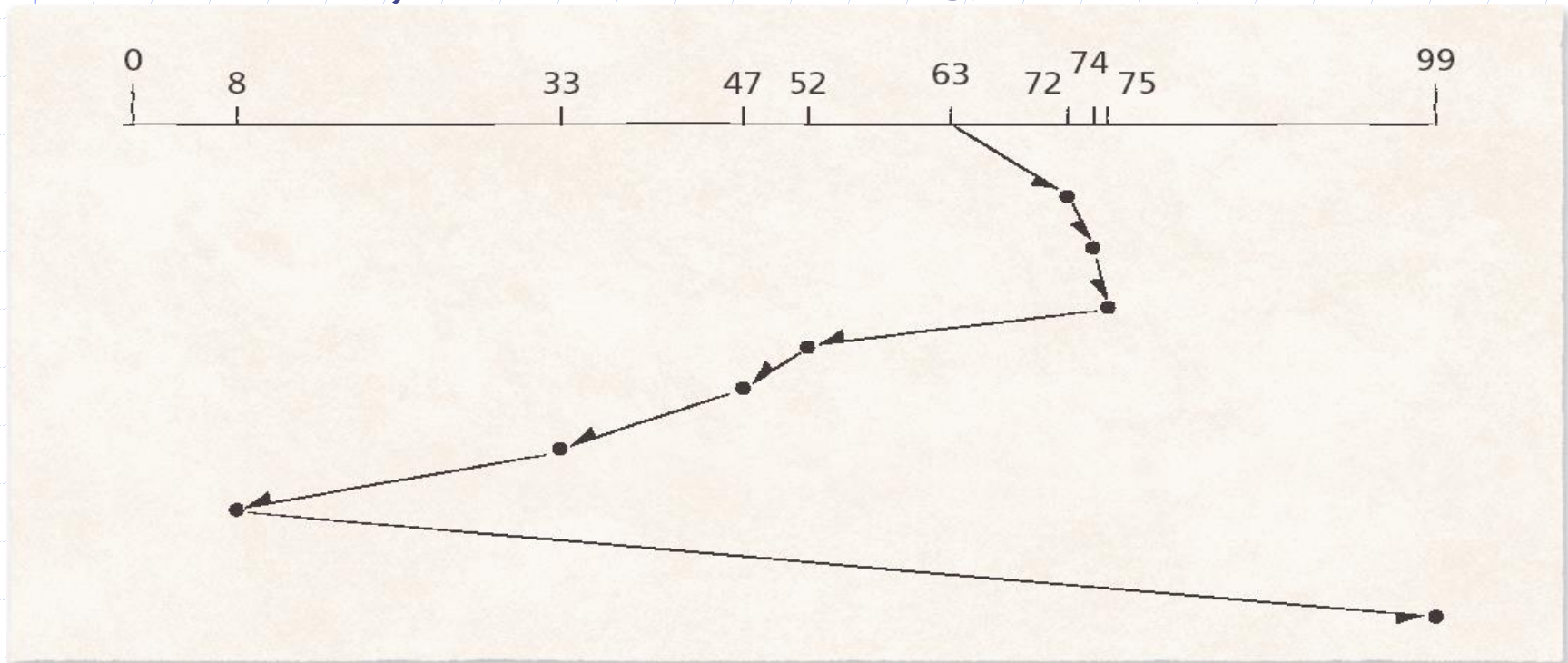
## ◆ ການຈັດຕາຕະລາງແບບໃຫ້ບໍລິການອັນທີ່ໃຊ້ເວລາເຄື່ອນຍ້າຍຫົວອ່ານນ້ອຍທີ່ສຸດ (SSTF Scheduling)

- ໃຫ້ບໍລິການອັນທີ່ຢູ່ໄກ່ຫົວອ່ານທີ່ສຸດ
- ຈຸດດີ
  - ◆ ສາມາດໃຫ້ບໍລິການໃນຫົວໜ່ວຍເວລາໄດ້ໜຶ່ງໄດ້ຫຼາຍຂຶ້ນ ແລະ ເວລາລໍຖ້າໂດຍສະເລ່ຍລຸດລົງເມື່ອທຽບກັບ ວິທີ FCFS
  - ◆ ເໝາະສົມໃຊ້ງານກັບລະບົບ Batch Processing System
- ຜົນເສຍ
  - ◆ ບໍ່ຮັບປະກັນຄວາມຍຸດຕິທຳ
  - ◆ ອາດຈະເຮັດໃຫ້ເກີດການເລື່ອນການເຮັດວຽກໄປຕະຫຼອດການ
  - ◆ ເວລາປ່ຽນແປງສູງ
  - ◆ ເວລາຕອບສະໜອງບໍ່ເປັນທີ່ຍອມຮັບໃນລະບົບໂຕ້ຕອບ



# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

- ການຈັດຕາຕະລາງແບບໃຫ້ບໍລິການອັນທີ່ໃຊ້ເວລາເຄື່ອນຍ້າຍຫົວອ່ານນ້ອຍທີ່ສຸດ (SSTF Scheduling)



Queue: 33, 72, 47, 8, 99, 74, 52, 75. Head position: 63

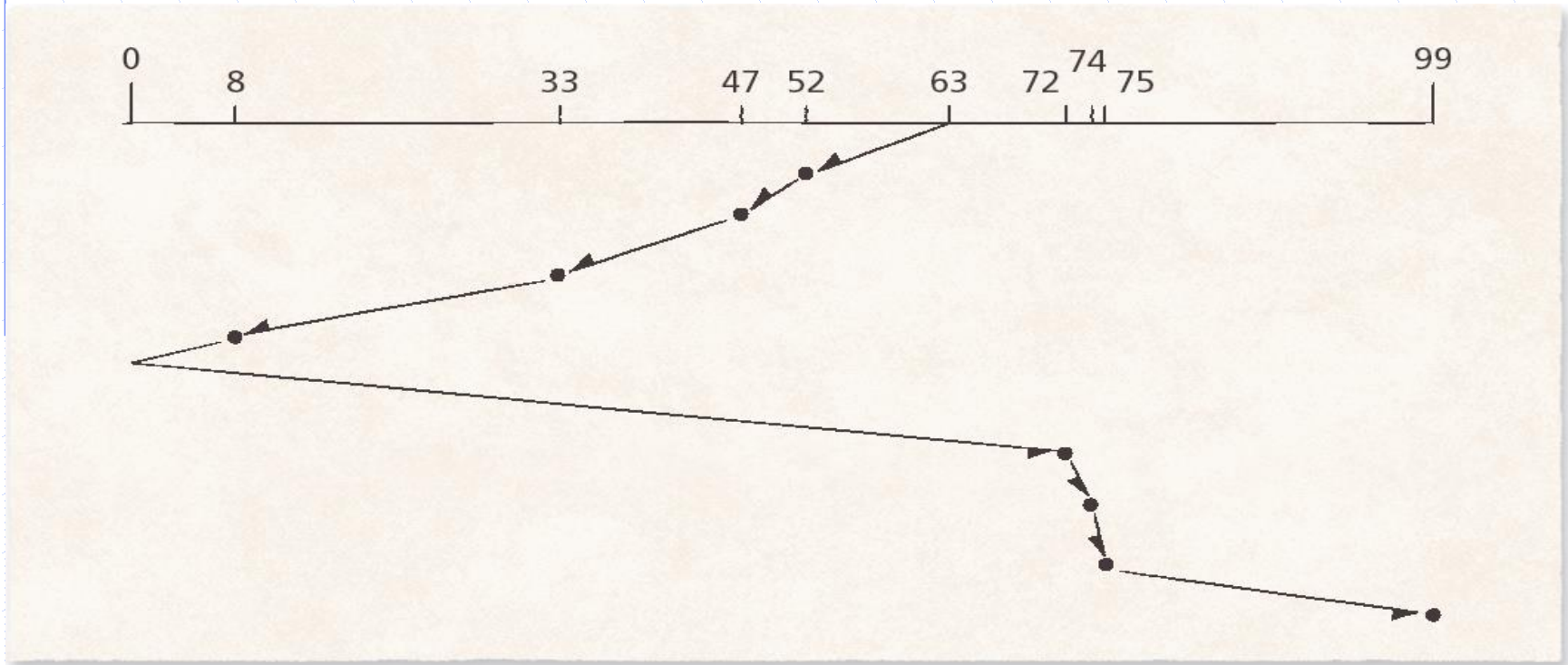
ຮູບແບບການໃຫ້ບໍລິການດ້ວຍວິທີການ SSTF

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

- ◆ **ການຈັດຕາຕະລາງແບບ SCAN (SCAN Disk Scheduling)**
  - ໃຫ້ບໍລິການອັນທີ່ຢູ່ໄກ້ຫົວອ່ານທີ່ສຸດຕາມທິດທາງໃດໜຶ່ງຈົນສຸດ
  - ມັນຈະບໍ່ປ່ຽນທິດທາງຈົນກ່ວາຫົວອ່ານໄປເຖິງດ້ານໃດໜຶ່ງ
  - ມີລັກສະນະການເຮັດວຽກຄ້າຍຄືກັບ SSTF
  - ອາດຈະເກີດການເລື່ອນການເຮັດວຽກໄປຕະຫຼອດການ
  - ເຮັດໃຫ້ເວລາປ່ຽນແປງດີຂຶ້ນ (ໃຊ້ເວລາປ່ຽນແປງໜ້ອຍລົງ)

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການຈັດຕາຕະລາງແບບ SCAN (SCAN Disk Scheduling)



Queue: 33, 72, 47, 8, 99, 74, 52, 75. Head position: 63

ຮູບແບບການໃຫ້ບໍລິການດ້ວຍວິທີການ SCAN

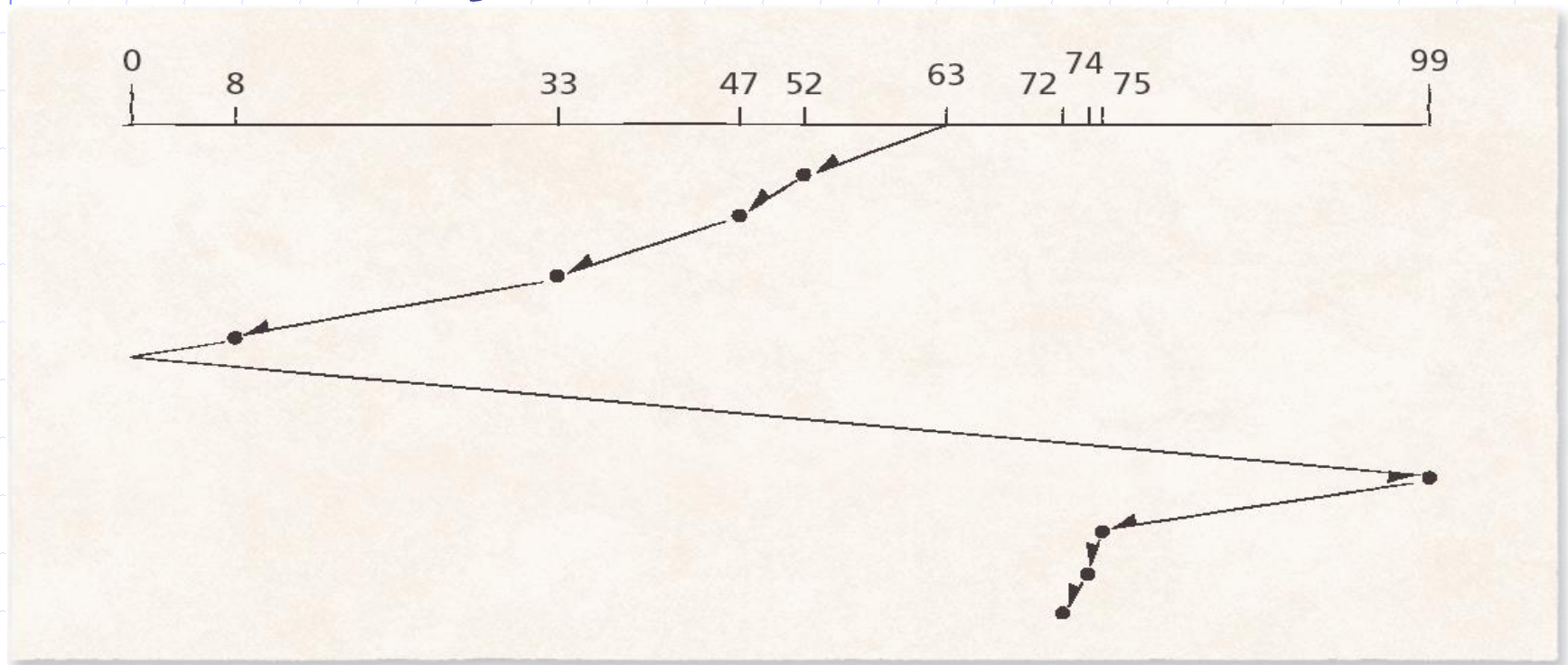
# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການຈັດຕາຕະລາງແບບ C-SCAN

- ການເຮັດວຽກຄ້າຍຄືກັບ SCAN, ແຕ່ຫຼັງຈາກທີ່ຫົວອ່ານໄປເຖິງຂອບດ້ານໃນແລ້ວ ມັນຈະເຕັ້ນກັບອອກຫາຂອບດ້ານນອກເລີຍ ໂດຍບໍ່ໃຫ້ບໍລິການໃດໆ
- ສາມາດລຸດເວລາປ່ຽນແປງ ແຕ່ການໃຫ້ບໍລິການຕໍ່ຫົວໜ່ວຍເວລາໄດ້ໜ້ອຍ ແລະ ເຮັດໃຫ້ເວລາໂດຍສະເລ່ຍເພີ່ມຂຶ້ນ

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການຈັດຕາຕະລາງແບບ C-SCAN



Queue: 33, 72, 47, 8, 99, 74, 52, 75. Head position: 63

ຮູບແບບການໃຫ້ບໍລິການດ້ວຍວິທີການ C-SCAN

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

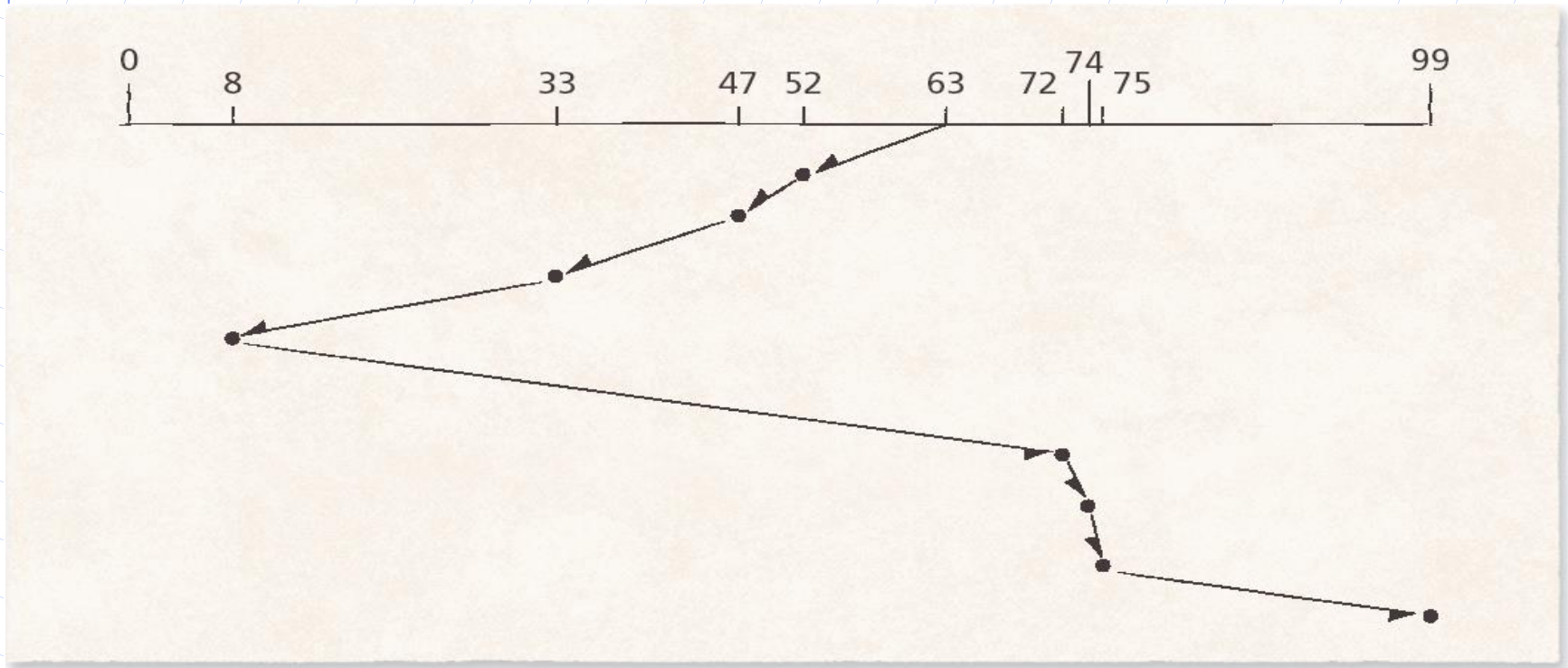
## ◆ ການຈັດຕາຕະລາງແບບ LOOK ແລະ C-LOOK

- LOOK: ເປັນວິທີການທີ່ປັບປຸງວິທີ SCAN
  - ◆ ມັນຈະເຄື່ອນຫົວອ່ານອອກໄປຫາຂອບນອກເລີຍຖ້າບໍ່ມີການການຂໍຮ້ອງໃນເຂດນັ້ນ
  - ◆ ເຮັດໃຫ້ປະສິດທິພາບດີຂຶ້ນ ໂດຍບໍ່ປະຕິບັດອັນທີ່ບໍ່ຈຳເປັນ
  - ◆ ເຮັດໃຫ້ສາມາດໃຫ້ບໍລິການໄດ້ຫຼາຍຂຶ້ນຕໍ່ຫົວໜ່ວຍເວລາ
- C-LOOK: ເປັນວິທີການທີ່ປັບປຸງວິທີ C-SCAN
  - ◆ ເປັນວິທີປະສົມປະສານວິທີ LOOK ແລະ C-SCAN
  - ◆ ເຮັດໃຫ້ເວລາປ່ຽນແປງນ້ອຍລົງເມື່ອທຽບກັບ LOOK, ແຕ່ການໃຫ້ບໍລິການໃນຫົວໜ່ວຍເວລາໃດໜຶ່ງໜ້ອຍກ່ວາ



# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ການຈັດຕາຕະລາງແບບ LOOK ແລະ C-LOOK



Queue: 33, 72, 47, 8, 99, 74, 52, 75. Head position: 63

ຮູບແບບການໃຫ້ບໍລິການດ້ວຍວິທີການ LOOK

# ຍຸດທະສາດໃນການຈັດຕາຕະລາງໃຫ້ແກ່ Disk

## ◆ ການເລືອກການຈັດຕາຕະລາງຂອງ Disk

- SSTF ເປັນວິທີທີ່ໃຊ້ຢູ່ທົ່ວໄປແລະມີຄວາມເປັນທຳມະຊາດເພາະວ່າສາມາດເພີ່ມປະສິດທິພາບໄດ້ດີກ່ວາແບບ FCFS
- ແບບ Scan ແລະ C-Scan ສາມາດຮອງຮັບການປະຕິບັດງານໄດ້ດີໃນສະພາບທີ່ມີການໃຊ້ງານ disk ຢ່າງໜັກ
- ປະສິດທິພາບຂອງຂັ້ນຕອນວິທີເລື່ອງນີ້ຂຶ້ນຢູ່ກັບ
  - ◆ ຈຳນວນແລະຊະນິດຂອງການຮ້ອງຂໍ
  - ◆ ວິທີຈັດສັນ File
  - ◆ ຕຳແໜ່ງຂອງ block ທີ່ເກັບ directory ແລະ ຕັດຊະນີ
- ລະບົບປະຕິບັດການຄວນມີຫລາຍທາງເລືອກໃນການໃຊ້ຂັ້ນຕອນວິທີການຈັດຕາຕະລາງ disk ເພື່ອປັບໃຫ້ເໝາະສົມກັບການຮ້ອງຂໍຮູບແບບຕ່າງໆ

# ສິ່ງທີ່ຄວນພິຈາລະນາ

- ◆ ການຈັດຕາຕະລາງໃຫ້ແກ່ Disk ຈະຊ່ວຍປະຢັດເວລາການອ່ານ/ຂຽນໄດ້ຫຼາຍ, ແຕ່ບໍ່ແມ່ນເປັນແນວນັ້ນທຸກກໍລະນີ
  - ມັນຈະບໍ່ຊ່ວຍໃນກໍລະນີລະບົບທີ່ໃຊ້ CPU ເປັນຫຼັກ
  - ຈະໃຫ້ປະໂຫຍດສູງສຸດໃນລະບົບທີ່ມີການໂລດ transaction ນ້ອຍໆ ເພື່ອກະຈາຍແຕ່ລະທີ່ຕັ້ງແບບສຸ່ມ
  - ການກະຈາຍທີ່ບໍ່ເປັນແບບສຸ່ມ ການເຮັດວຽກຂອງ Algorithm ຈະເຮັດໃຫ້ປະສິດທິພາບຂອງລະບົບລຸດລົງ
  - ບາງຄັ້ງວິທີການຈັດການ File ຈະເຮັດໃຫ້ປະສິດທິພາບການເຮັດວຽກຂອງ Algorithm ລຸດລົງ

# Caching ແລະ Buffering

- ◆ Cache ແລະ buffer: ແມ່ນການເກັບຊຸດສໍານຶກຂໍ້ມູນຂອງ disk ໄວ້ໃນໜ່ວຍຄວາມຈໍາທີ່ມີຄວາມໄວສູງ
  - ເກັບໄວ້ໃນໜ່ວຍຄວາມຈໍາຫຼັກ, ໃນ Cache ຫຼື ໃນຕົວຄວບຄຸມດິດ
  - ເຮັດໃຫ້ການເຂົ້າໄປຫາຂໍ້ມູນໄວກວ່າເຂົ້າໄປໃນດິດ
  - ສາມາດໃຊ້ເປັນບ່ອນເກັບຂໍ້ມູນຊົ່ວຄາວເພື່ອລໍຖ້າຂຽນຂໍ້ມູນຈົນກະທັ້ງດິດມີວຽກໜ້ອຍລົງ
- ◆ ອາດຈະເຮັດໃຫ້ເກີດການບໍ່ຊອດຄ່ອງກັນ
  - ຂໍ້ມູນຢູ່ໃນໜ່ວຍຄວາມຈໍາຫຼັກອາດຈະສຸນເສຍໃນກໍລະນີແຮງໄຟຟ້າເກີນ ຫຼື ລະບົບລົ້ມ
  - Write-back caching ຂໍ້ມູນຈະບໍ່ຖືກຂຽນລົງໃນດິດໂດຍທັນທີ
  - Write-through caching ຂຽນຂໍ້ມູນລົງໃນດິດ ແລະ cache ພ້ອມກັນ

# ການປັບປະສິດທິພາບຂອງ Disk ແບບອື່ນໆ

## ◆ ການເຮັດໃຫ້ປະສິດທິພາບດີຂຶ້ນນັ້ນຍັງມີວິທີອື່ນໆດັ່ງນີ້:

- ການຈັດສັນຂໍ້ມູນຄືນໃໝ່ (Defragmentation?)
  - ◆ ເປັນການເອົາຂໍ້ມູນທີ່ພົວພັນກັນມາເກັບໄວ້ຕໍ່ເນື່ອງກັນ
  - ◆ ເຮັດໃຫ້ລຸດເວລາການຄົ້ນຫາ
  - ◆ ການແບ່ງສ່ວນອອກເຮັດໃຫ້ລຸດການເກີດເນື້ອທີ່ວ່າງທີ່ໃຊ້ບໍ່ໄດ້
- ການບີບອັດຂໍ້ມູນ (Compression)
  - ◆ ແມ່ນການເຮັດໃຫ້ປະຢັດເນື້ອທີ່ດິດໃນການເກັບຂໍ້ມູນ
  - ◆ ເຮັດໃຫ້ເວລາໃນການເຂົ້າຫາຂໍ້ມູນ ແລະ ເວລາໃນການສົ່ງຂໍ້ມູນດີຂຶ້ນ
  - ◆ ເຮັດໃຫ້ເວລາໃນການເຮັດວຽກເພີ່ມຂຶ້ນ ເພາະວ່າຕ້ອງໄດ້ໃຊ້ເວລາໃນການບີບອັດ ແລະ ເວລາໃນການແຕກຂໍ້ມູນອອກມາ

# ການປັບປະສິດທິພາບຂອງ Disk ແບບອື່ນໆ

## ◆ ການເຮັດໃຫ້ປະສິດທິພາບດີຂຶ້ນນັ້ນຍັງມີວິທີອື່ນໆດັ່ງນີ້:

- ສໍາເນົາຊຸດຂໍ້ມູນທີ່ເອີ້ນໃຊ້ເປັນປະຈຳອອກເປັນຫຼາຍຊຸດ
  - ◆ ສາມາດເຂົ້າໄປໃຊ້ຊຸດຂໍ້ມູນທີ່ຢູ່ໄກຫົວອ່ານ/ຂຽນທີ່ສຸດ
  - ◆ ເຮັດໃຫ້ເປືອງເນື້ອທີ່ໃນການເກັບ
- ບັນທຶກຂໍ້ມູນເທື່ອລະກ້ອນ (Record Blocking)
  - ◆ ແມ່ນການອ່ານ/ຂຽນຂໍ້ມູນຫຼາຍຊຸດພ້ອມກັນເປັນເທື່ອລະກ້ອນ
- ການວາງຕໍາແໜ່ງຂອງແຂນດິດ
  - ◆ ເມື່ອຈະຢຸດການເຮັດວຽກ ໃຫ້ວາງແຂນດິດໄປໄວ້ໄກກັບຂໍ້ມູນທີ່ຄາດວ່າຈະຖືກອ່ານຕໍ່ໄປ
  - ◆ ຖ້າການຄາດຄະເນເບບໍ່ຖືກຕ້ອງ ຈະເຮັດໃຫ້ປະສິດທິພາບລະບົບລຸດລົງ



# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## ◆ ມີຫຼາຍເທັກນິກທີ່ໃຊ້ໃນການປັບປຸງ Disk ເຊັ່ນ:

- ການໃຊ້ disk ຫຼາຍຕົວເຮັດວຽກຊ່ວຍກັນເພື່ອປັບປຸງຄວາມໄວ
- ການເກັບຂໍ້ມູນຢາຍໄວ້ຫລາຍ disk
- ການເອົາໂຮມເອົາເນື້ອທີ່ຈາກຫລາຍ disk ເຂົ້າເປັນອັນດຽວ

## ◆ ແນວຄິດດັ່ງກ່າວແມ່ນການສ້າງ Disk Array

## ◆ Disk Array

- ແມ່ນການເອົາ Hard disk ຫລາຍຕົວມາຕໍ່ລວມເຂົ້າກັນ ເຮັດໃຫ້ເຫັນ Logical Drive ເໝືອນດັ່ງເປັນ Disk ຕົວດຽວທີ່ມີເນື້ອທີ່ບັນຈຸເພີ່ມຂຶ້ນ ເພື່ອໃຫ້ສາມາດເຮັດວຽກໄດ້ຢ່າງມີປະສິດທິພາບ, ເປັນລະບົບທຶນທານຕໍ່ຄວາມຜິດພາດ ຊຶ່ງເອີ້ນວ່າ RAID (Redundant Array of Inexpensive Disk)

# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## ◆ RAID Technology

- ເປັນເທັກໂນໂລຍີທີ່ໃຊ້ສ້າງ Disk Array
- ສາມາດເຮັດໄດ້ທັງ Hardware RAID ແລະ Software RAID
- ໃຊ້ Hard disk ຢ່າງໜ້ອຍ 2 ຕົວຂຶ້ນໄປ

# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## ◆ ປະໂຫຍດຂອງລະບົບ RAID

- **Reliability** (ຄວາມໜ້າເຊື່ອຖື) ມີລະບົບ Fault Tolerance/ Data Redundancy ເພື່ອສ້າງຄວາມໝັ້ນຄົງ ແລະ ປອດໄພໃຫ້ແກ່ຂໍ້ມູນ
- **Performance** (ປະສິດທິພາບ) ມີຄວາມໄວໃນການສົ່ງຂໍ້ມູນເພີ່ມຂຶ້ນ ຖ້າທຽບກັບການໃຊ້ hard disk ຕົວດຽວ
- **Capacity** (ເນື້ອທີ່ບັນຈຸຂໍ້ມູນ) ສາມາດເພີ່ມເນື້ອທີ່ໃນການບັນຈຸຂໍ້ມູນ
- **Data Backup** (ເກັບສຳລອງຂໍ້ມູນ) ເປັນລະບົບເກັບຂໍ້ມູນສຳລອງໄວ້

# ຄວາມໝ້າເຊື່ອຖືຂອງ Disk

## ◆ ລະດັບຂອງ RAID

- ລະດັບ 0 (Level 0) Disk Striping
- ລະດັບ 1 (Level 1) Disk Mirroring
- ລະດັບ 3 (Level 3) Disk Striping with a dedicate parity disk
- ລະດັບ 5 (Level 5) Disk Striping with parity distributed across multiple drive
- ລະດັບ 0+1 (Level 0+1) Disk Striping & Mirroring
- ລະດັບ 30 (Level 30) Disk Striping of dedicate parity Array
- ລະດັບ 50 (Level 50) Disk Striping of distribute parity Array

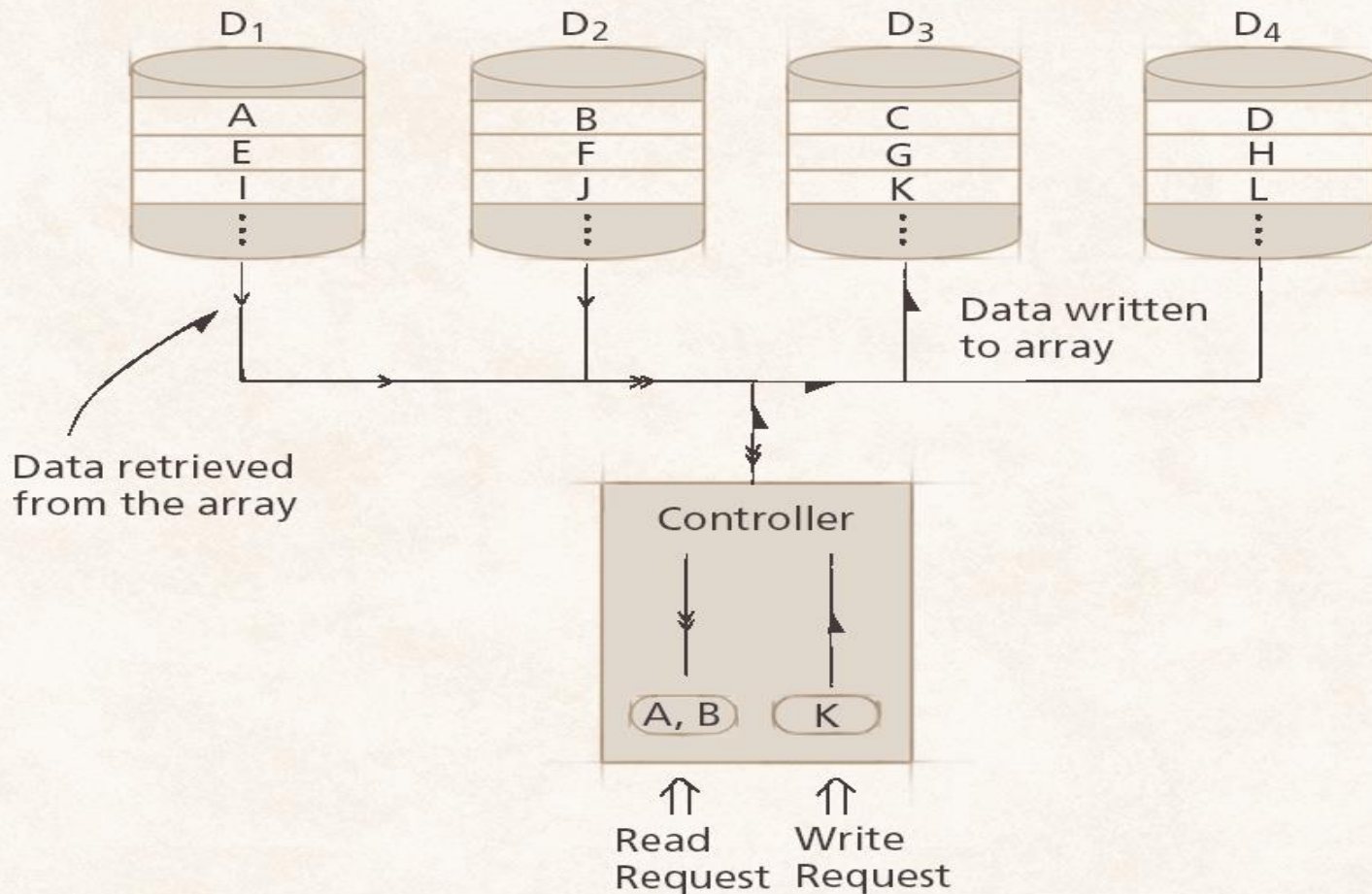
# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## □ RAID ລະດັບ 0 (Disk Striping)

- ເປັນການເອົາ Hard Disk ຫລາຍໆຕົວມາຕໍ່ລວມເຂົ້າກັນ ໃຫ້ເບິ່ງເຫັນເນື້ອທີ່ທັງໝົດເປັນ Hard Disk ໃຫຍ່ຕົວໜຶ່ງ
- ຫລັກການເຮັດວຽກຂອງ RAID 0 ແມ່ນແບ່ງ File ອອກໄປເກັບໄວ້ໃນແຕ່ລະ Block ຂອງ Hard Disk ແຕ່ລະຕົວແບບກະຈາຍໄປທຸກໆຕົວ
- ການອ່ານ, ຂຽນຂໍ້ມູນເຮັດໄດ້ໄວ
- ແຕ່ບໍ່ມີລະບົບປ້ອງກັນຄວາມຜິດພາດ
- ເນື້ອທີ່ລວມທັງໝົດເທົ່າກັບເນື້ອທີ່ໃນ Hard Disk ທີ່ມີຂະໜາດນ້ອຍທີ່ສຸດຄຸນດ້ວຍຈຳນວນ Hard Disk ທັງໝົດ

# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## RAID ລະດັບ 0 (Disk Striping)

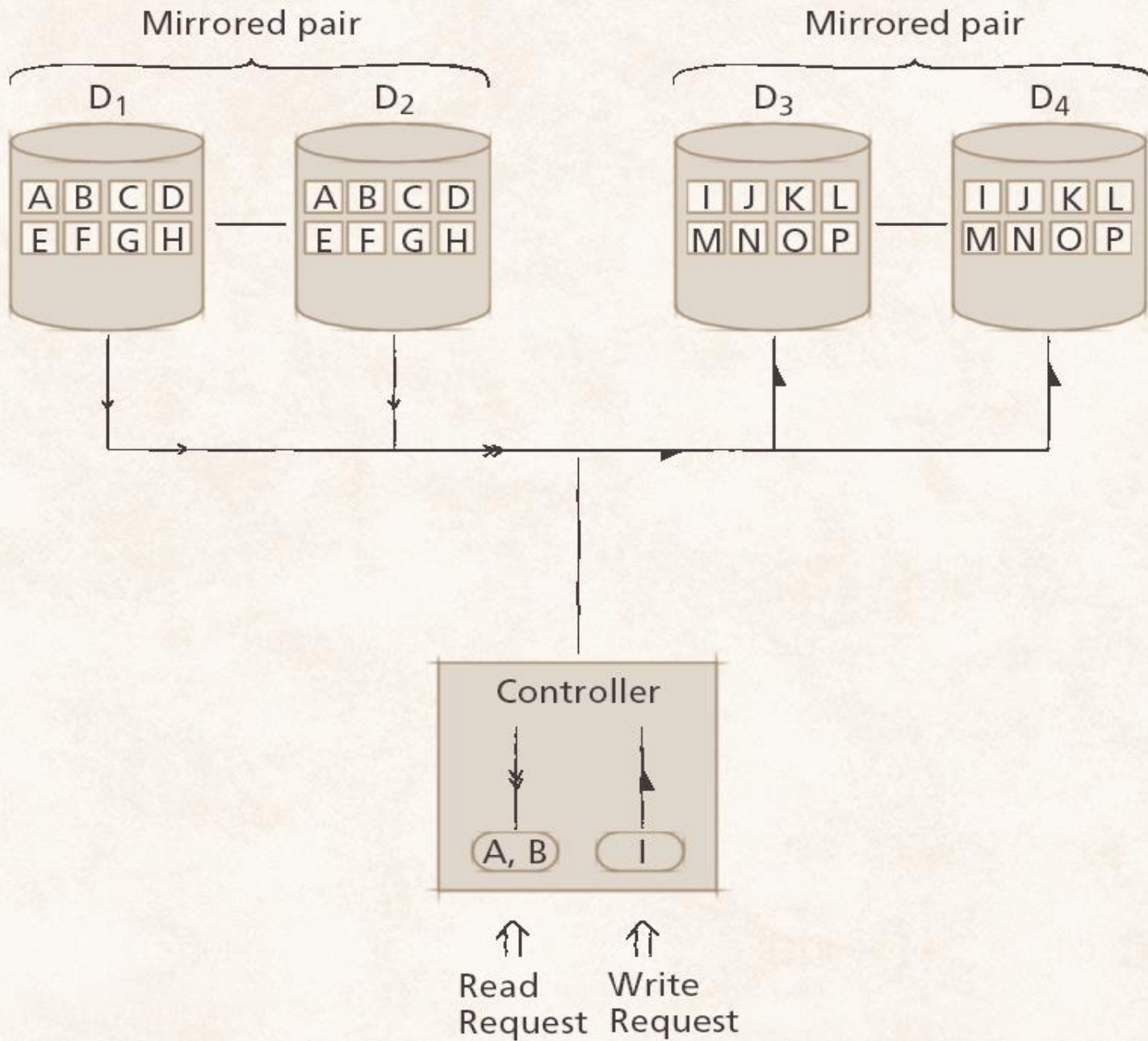




# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## ❑ RAID ລະດັບ 1 (Disk Mirroring)

- ເປັນການເອົາ Hard Disk ມາຕໍ່ເຂົ້າກັນເປັນຄູ່ໂດຍມີການເກັບຂໍ້ມູນແຕ່ລະອັນເປັນ 2 ຊຸດ, ເກັບແຕ່ລະຊຸດໄວ້ໃນ Disk ແຕ່ລະຕົວ
- ມີປະສິດທິພາບໃນການອ່ານຂໍ້ມູນ
- ແຕ່ສິ້ນເປືອງ ເນື້ອທີ່ Hard Disk (ເສຍ 50%)
- ເນື້ອທີ່ລວມທັງໝົດເທົ່າກັບເນື້ອທີ່ຂອງ Hard Disk ທີ່ນ້ອຍທີ່ສຸດຄຸນດ້ວຍຈຳນວນ Hard Disk ແລ້ວຫານໃຫ້ 2
- ເພື່ອເພີ່ມປະສິດທິພາບໃນການເຮັດວຽກ ແລະ ເພີ່ມລະບົບ Fault Tolerance, ຄວນຕິດຕັ້ງ Controller 2 ຕົວ

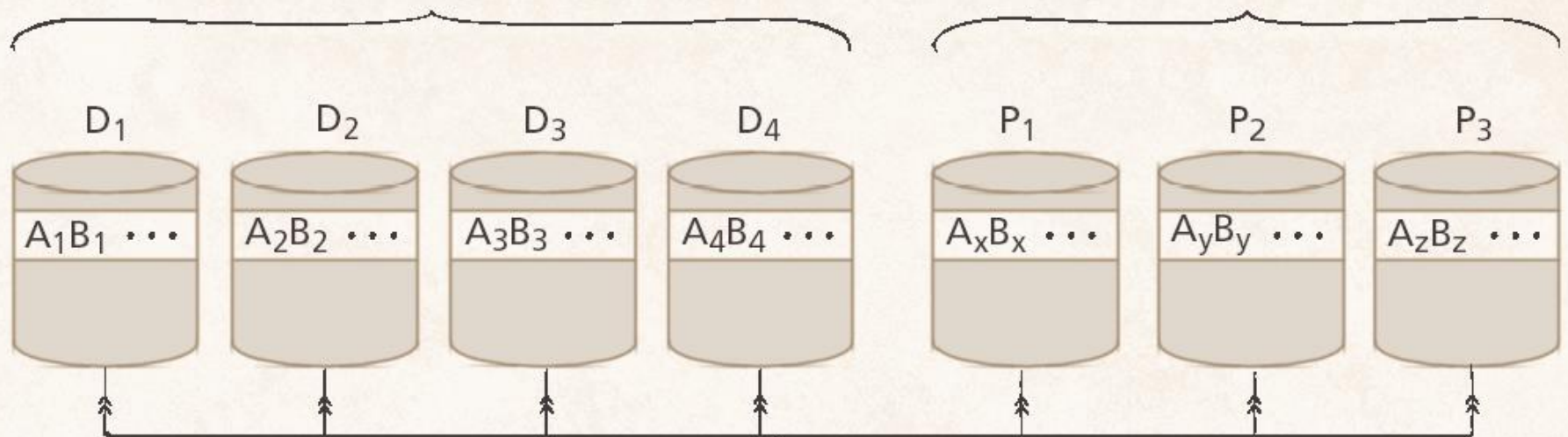


# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

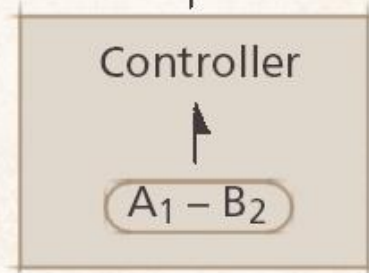
- ❑ RAID ລະດັບ 2 (Bit-Level Hamming ECC Parity)
  - ເປັນການເອົາ Hard Disk ຫຼາຍໂຕມາຕໍ່ເຂົ້າກັນໂດຍທີ່ມີການເກັບຂໍ້ມູນຢາຍໃສ່ແຕ່ລະອັນ ຊຶ່ງຂໍ້ມູນຈະຖືກຕັດອອກເປັນຫຼາຍໆ bit
  - ໃຊ້ Hamming ECC ເພື່ອກວດສອບຄວາມຖືກຕ້ອງສົມບູນຂອງຂໍ້ມູນ ຊຶ່ງມີການເກັບ parity ຢາຍໄວ້ຫຼາຍ disk
  - ບໍ່ມີການຜະລິດຂຶ້ນມາໃຊ້ຈິງ

Data disks  
(stores data strips)

Parity disks  
(stores parity strips)



Controller must read  
from all data disks in  
the array and write to  
all parity disks to  
perform the write.



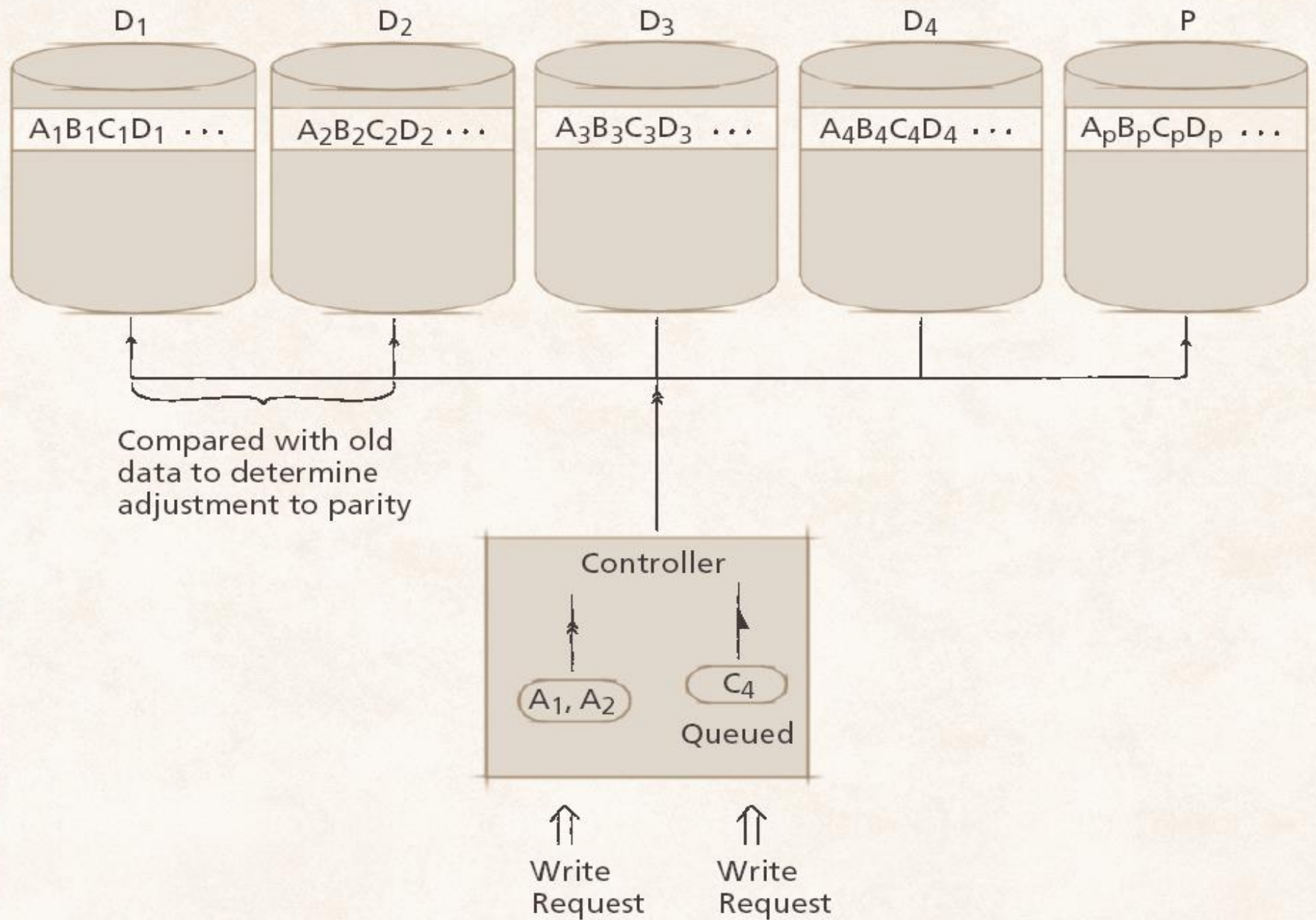
Write request

# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

## □ RAID ລະດັບ 3 (Bit-Level XOR ECC Parity)

- ວິທີການຕໍ່ ແລະ ການເກັບຂໍ້ມູນຄືກັບ RAID 2, ແ ມີ Hard Disk ຢ່າງໜ້ອຍ 3 ຕົວ, 2 ຕົວແມ່ນເອົາໄວ້ເກັບຂໍ້ ມູນ ແລະ ອີກໜຶ່ງຕົວແມ່ນໄວ້ເກັບ Parity
- ໃຊ້ Function XOR ໃນການຄຳນວນຂໍ້ມູນສຳຫລັບສ້າງ Parity
- ເນື້ອທີ່ລວມຈະເທົ່າກັບຈຳນວນ Hard Disk ທັງໝົດລົບ ອອກ 1 ແລ້ວຄູນດ້ວຍເນື້ອທີ່ຂອງ Hard Disk ທີ່ໜ້ອຍສຸດ
- ບໍ່ເໝາະສົມກັບວຽກທີ່ມີການ Random ເປັນປະຈຳ ຊຶ່ງເຮັດ ໃຫ້ Parity Disk ຄຳນວນເປັນປະຈຳ ແລ້ວເຮັດໃຫ້ປະສິດທິ ພາບຕ່ຳ





# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

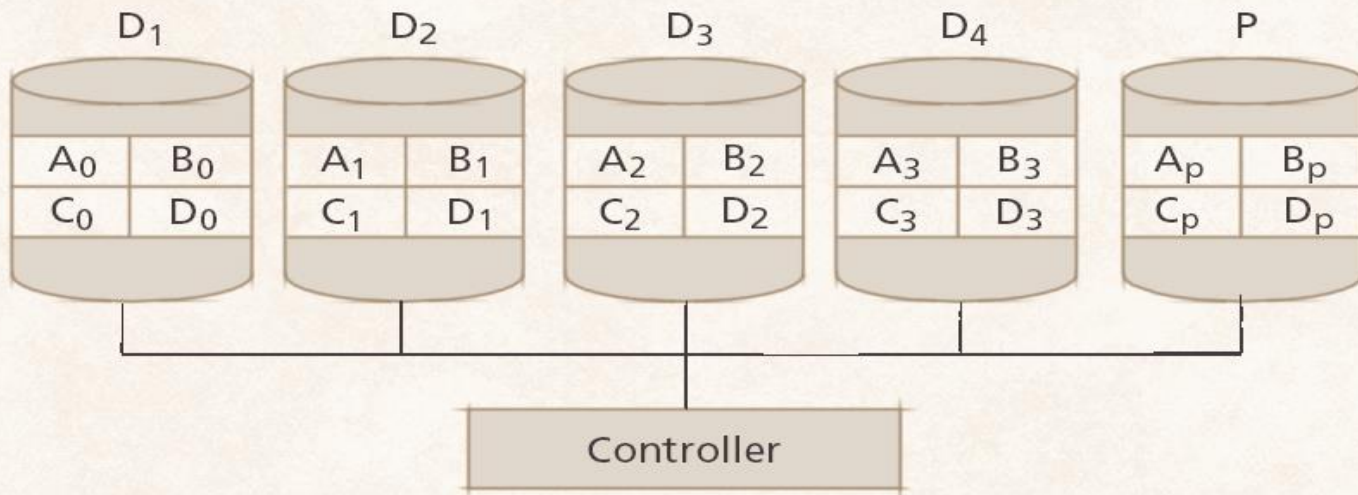
## ❑ RAID ລະດັບ 4 (Block-Level XOR ECC Parity)

- ວິທີການຕໍ່ ແລະ ເກັບຂໍ້ມູນຄືກັນກັບ RAID 3, ແຕ່ການແບ່ງຂໍ້ມູນຈະເປັນ Block
- ໃຊ້ Function XOR ໃນການຄຳນວນຂໍ້ມູນສຳຫຼັບສ້າງ Parity
- ເນື້ອທີ່ລວມຈະເທົ່າກັບຈຳນວນ Hard Disk ທັງໝົດລົບ ອອກ 1 ແລ້ວຄູນດ້ວຍເນື້ອທີ່ຂອງ Hard Disk ທີ່ໜ້ອຍສຸດ
- ບໍ່ມີການຜະລິດໃຊ້ງານຈິງ ເພາະວ່າ ລະດັບທີ່ 5 ທີ່ຄ້າຍຄືກັນມີປະສິດທິພາບທີ່ດີກ່ວາ



# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

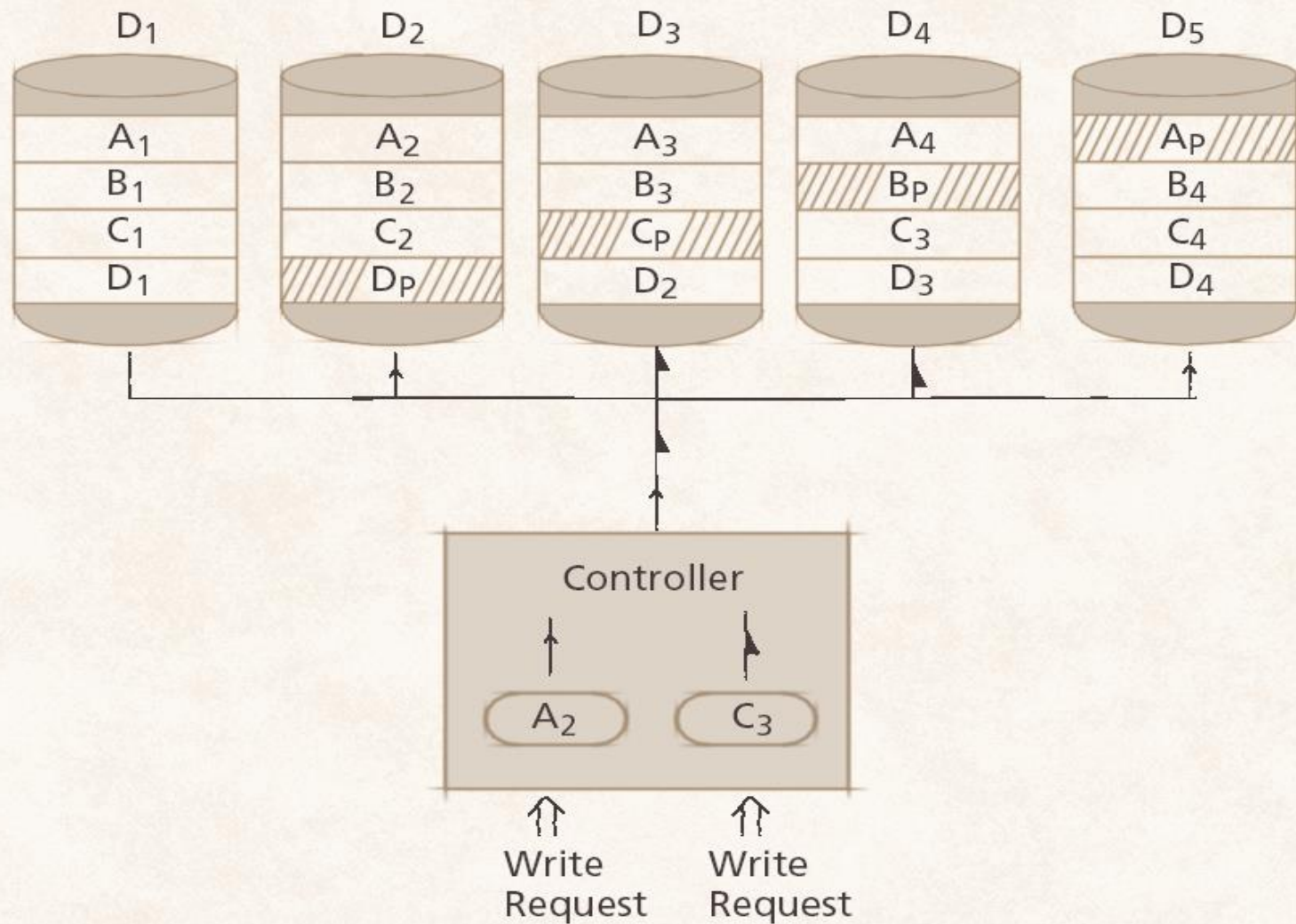
## ❑ RAID ລະດັບ 4 (Block-Level XOR ECC Parity)



# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk

- ❑ RAID ລະດັບ 5 (Block-Level Distributed XOR ECC)
  - ການເກັບຂໍ້ມູນຄ້າຍຄືກັບ RAID 4 ທີ່ກະຈາຍຂໍ້ມູນໄປເກັບໃນ Hard Disk ທຸກຕົວ ແຕ່ຈະເກັບ parity ແບບ rotation ໄວ້ໃນ Hard Disk ທຸກຕົວ
  - ສາມາດແກ້ບັນຫາ Bottle neck ທີ່ເກີດຂຶ້ນໃນ RAID 3
  - ຈະຕ້ອງມີ Hard Disk ຢ່າງໜ້ອຍ 3 ຕົວ
  - ເນື້ອທີ່ລວມຈະເທົ່າກັບຈຳນວນ Hard Disk ທັງໝົດລົບ ອອກ 1 ແລ້ວຄູນດ້ວຍເນື້ອທີ່ຂອງ Hard Disk ທີ່ໜ້ອຍສຸດ
  - ລະບົບປ້ອງກັນຄວາມຜິດພາດ ແລະ ປະສິດທິພາບຄືກັບ RAID 3, 4 ແຕ່ການຂຽນຂໍ້ມູນຈະຊ້າກ່ວາ

# ຄວາມໜ້າເຊື່ອຖືຂອງ Disk



# ລະບົບປະຕິບັດການກັບ Disk

- ❑ ການດໍາເນີນງານຂອງລະບົບປະຕິບັດການ
  - ຈັດການອຸປະກອນທາງດ້ານກາຍຍະພາບສໍາຫຼັບໂປຣແກຣມປະຍຸກຕ່າງໆທີ່ຮ້ອງຂໍ
- ❑ ການຕິດຕັ້ງ Disk
  - ການເຂົ້າຫາຂໍ້ມູນໃນ disk ສາມາດເຂົ້າໄດ້ 2 ທາງ
    - ຜ່ານທາງ port ສົ່ງຂໍ້ມູນໄປຫາໜ່ວຍເກັບທີ່ຕໍ່ໂດຍກົງກັບເຄື່ອງຄອມພິວເຕີ
    - ຜ່ານທາງໜ່ວຍເກັບຂໍ້ມູນທີ່ຕໍ່ຢູ່ກັບເຄືອຂ່າຍ

# ລະບົບປະຕິບັດການກັບ Disk

## □ ໜ່ວຍເກັບທີ່ຕິດຕັ້ງເຂົ້າກັບເຄື່ອງຄອມພິວເຕີໂດຍກົງ

- ມີ port ຫລາຍປະເພດເຊັ່ນ
  - IDE, EIDE, ATA, SATA ສໍາຫລັບ PC ທົ່ວໄປ
  - SCSI, FC ສໍາຫລັບຄອມພິວເຕີທີ່ມີສະມັດຖະນະສູງ

## ■ ໜ່ວຍເກັບທີ່ຕິດຕັ້ງເຂົ້າກັບລະບົບເຄືອຂ່າຍ (NAS)

- ເປັນການເຂົ້າຫາໜ່ວຍເກັບຂໍ້ມູນຜ່ານທາງລະບົບເຄືອຂ່າຍ
- ເຄື່ອງ client ສາມາດໃຊ້ຊຸດຄໍາສັ່ງໄລຍະໄກ (RPC) ເພື່ອເອີ້ນໃຊ້ຂໍ້ມູນຈາກເຄືອຂ່າຍໄດ້
- NAS ເປັນວິທີໜຶ່ງທີ່ອໍານວຍຄວາມສະດວກໃນການໃຫ້ຄອມພິວເຕີໃນລະບົບເຄືອຂ່າຍໃຊ້ຂໍ້ມູນຮ່ວມກັນໄດ້