



# Introduction to Journaling

Chad Olstad

Larry Youngren

[Journal\\_Guru@Yahoo.com](mailto:Journal_Guru@Yahoo.com)



## The Journal Roadmap (other Journal sessions)

( 9:30 Sun) Introduction to Journaling



(8:00 Mon) Putting your Journal on a diet

(11:00 Mon) Remote Journal – It just keeps getting better

(8:00 Tues) Don't stub your toe in a journal intensive HA environment

(3:30 Tues) **Open LABs:**

- Try out newest Journal features
- Try out tools to analyze Journal behavior
- End to end replication on a shoestring
- Dig into High Availability journal enablers

## *The basics*

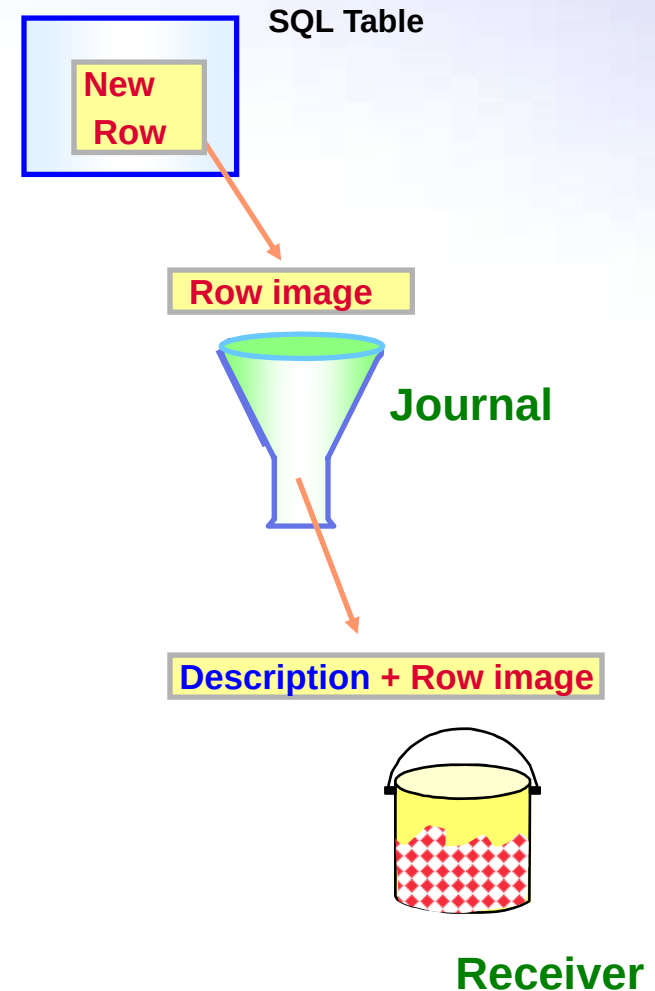
# Our building blocks

## Journal Object

- Holds information pertinent to journaling
- **Acts as a "funnel"**
- Ties together DB files and journal entries
- Does not grow in size over time

## Journal Receiver

- Houses the audit trail
- **Contains DB record images**
- Reveals: Who, what, when information
- Grows dramatically in size



# What Information is Captured?

- The variety of information identifying changes includes:

- **What**

- Image of database record, plus flavor of journal entry

- **When**

- Date, time, sequence

- **Who**

- User, job, program

- From **Where**

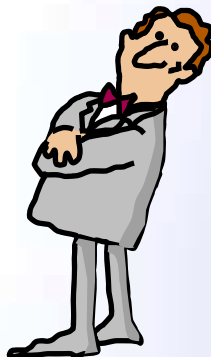
- IP Address

**Required**

**Default but optional**

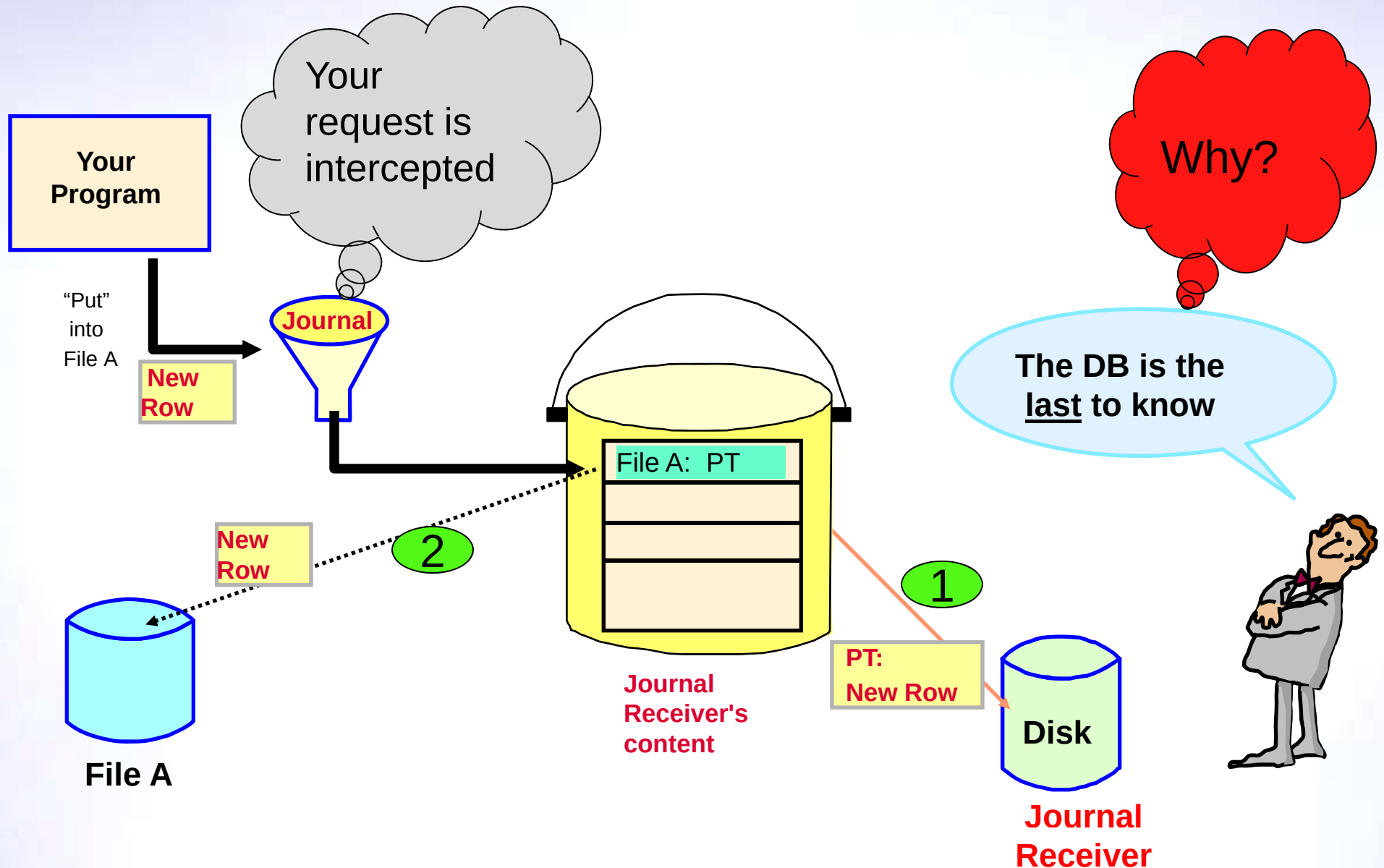
**Only if requested**

Journal knows more  
about your actions than  
the FBI...



# Simple Journaling example

## (What goes on when journaling is present)



# What Does Journaling Provide?

- Complete **audit trail** of DataBase transaction history
  - Produced automatically
    - ▶ No application changes required
  - In sequence - - it knows the order of operations
  - Regardless of source of changes
    - ▶ Across all jobs
  - Captured before DataBase updates are written
  - Securely
    - ▶ Can't be subsequently altered

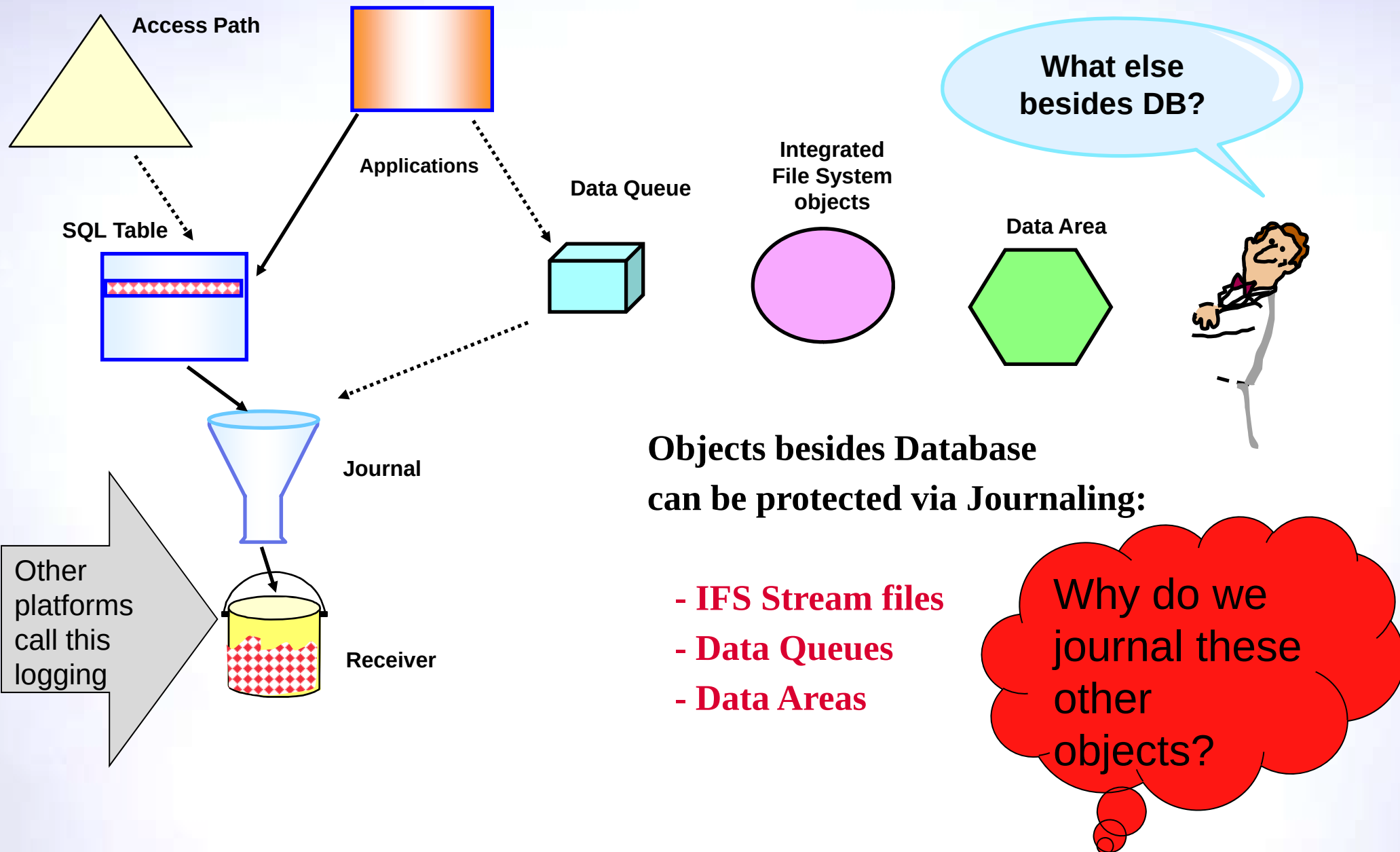
## Are you a Journal user ?

### ■ Yes, if...

- ▶ You employ SQL
- ▶ Any software packages you purchased employ SQL
  - Such as popular ERP packages
- ▶ You use Referential Integrity within your Database files
- ▶ You use commitment control
- ▶ You've got hefty sized Indexes (access paths)
  - ▶ And have not disabled System Managed Access Path Protection
- ▶ You employ an HA vendor replication package

***It's broader than merely database***

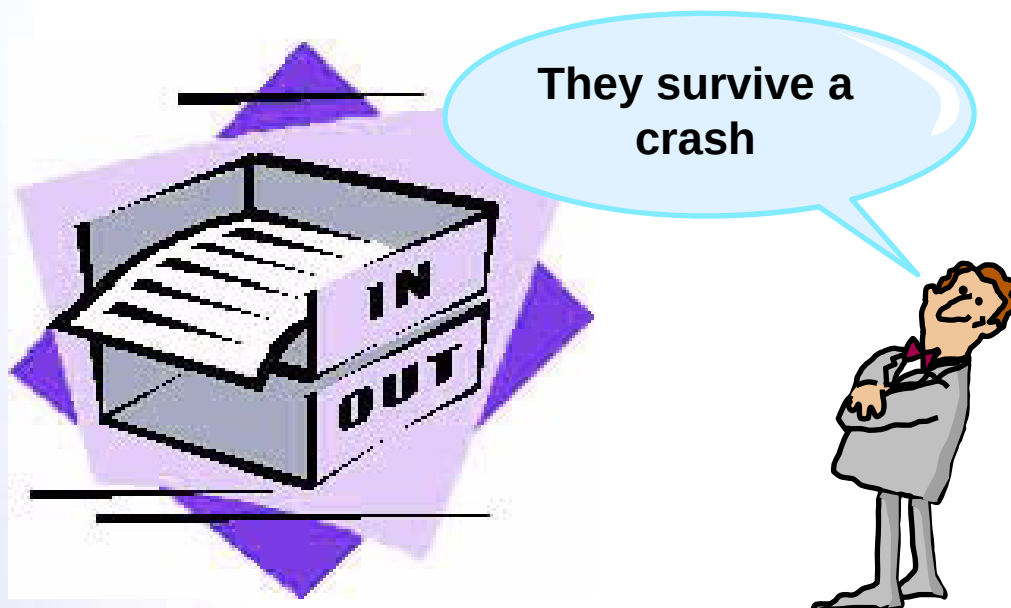
# Unlike other platforms, on i, it's not just a database enabler



# Journaling Data Queues

Data Queues - - why should I Journal them?

- Data queue Journaling affords **replication** in a HA environment
- Journaling of Data Queues provides robust **recovery during IPL.**
  - May help reduce likelihood of damage



A Fragile Object



When disease, disaster, and pestilence strike

# What other benefits does Journal provide?

Also assures faster recovery after crash:

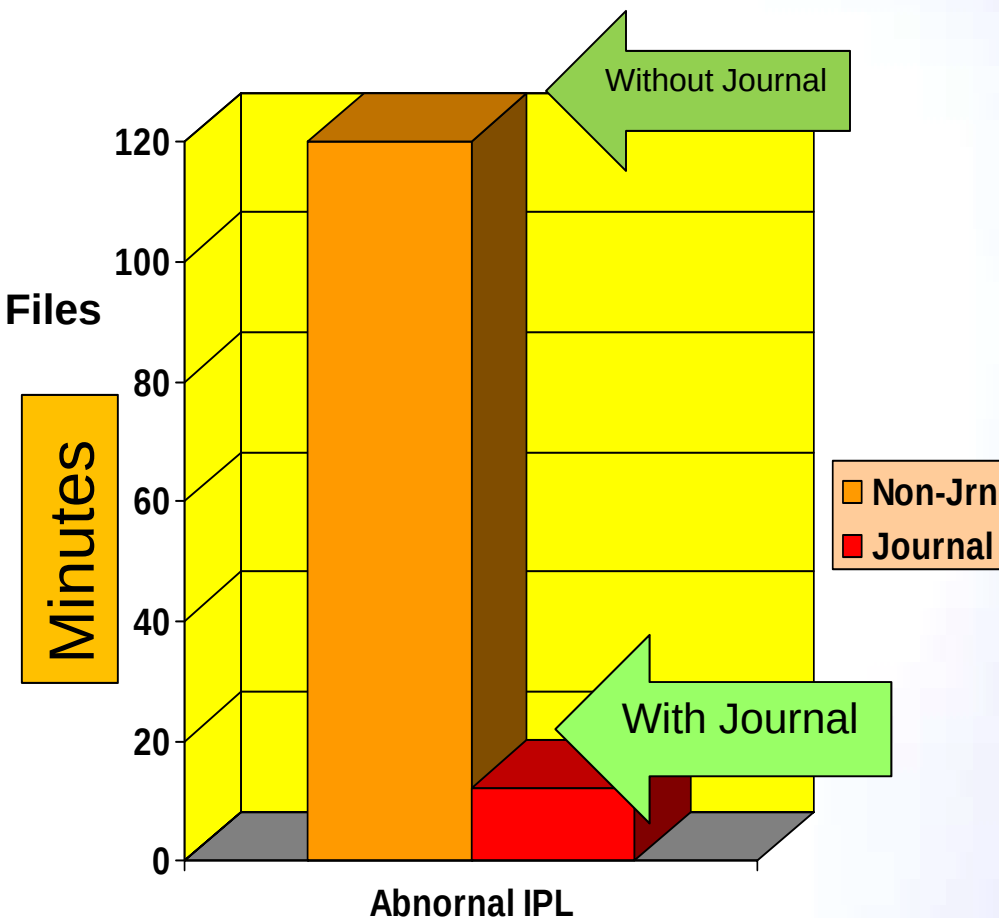
## Speeds up IPL:

Decreases recovery time of Access Paths significantly Hours drop to mere minutes

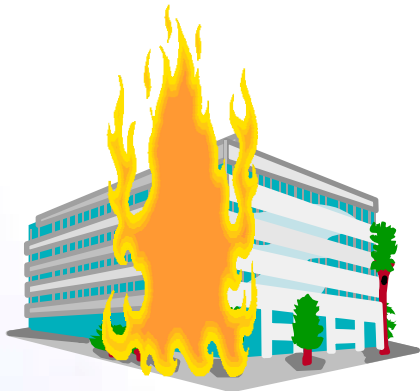
Decreases recovery time for certain Physical Files

PF using **REUSEDLT(\*YES)**

PF housing **variable length** fields



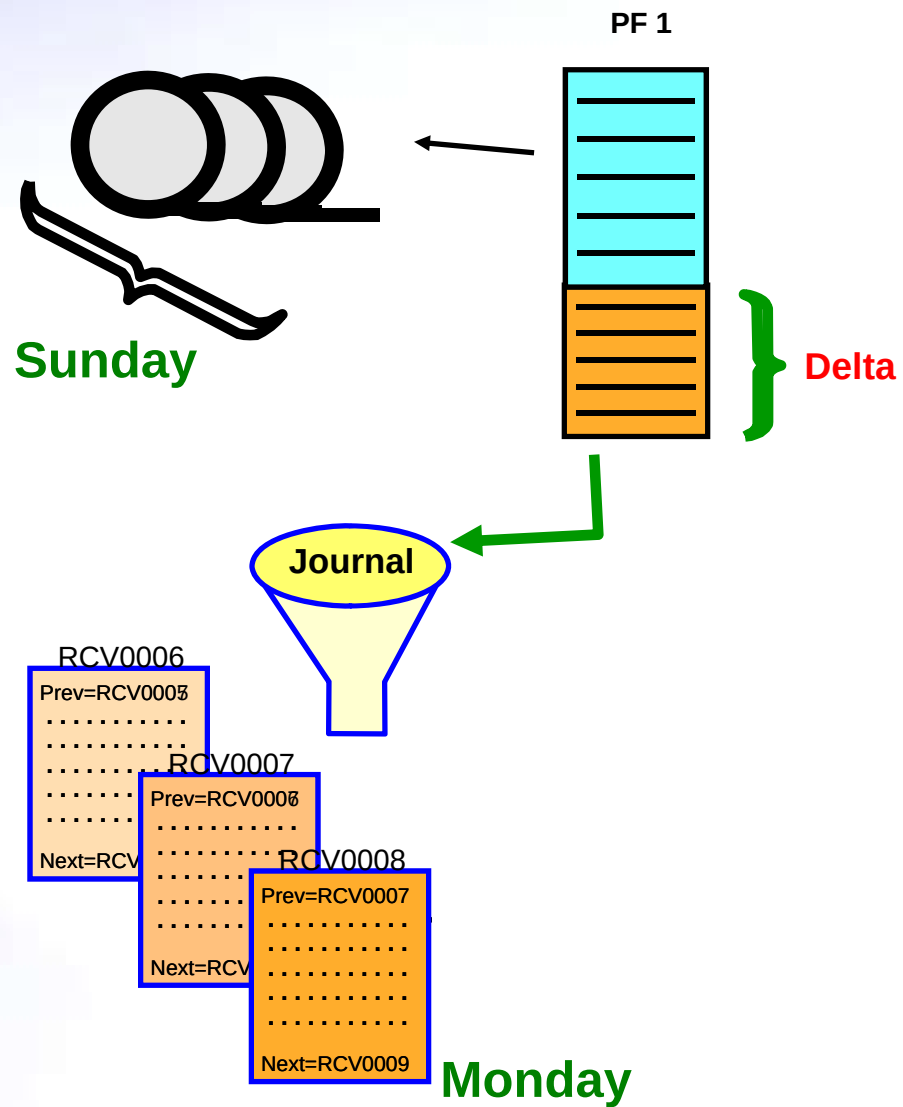
*The journal also helps you be prepared for site loss*



Data Center

# The daily save approach...

(Followed by recovery at a Hot site)



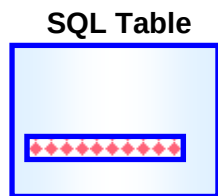
## Trimming nightly save windows

Perhaps you'll want to leverage journal to save less frequently

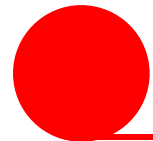
- With Journal you could save only today's updated/added records (Delta)
- Lets your nightly save finish sooner..
- Nightly Backup time and quantity of media can be reduced



1 Million



Receiver



2 Thousand

You can also  
customize your  
save strategy

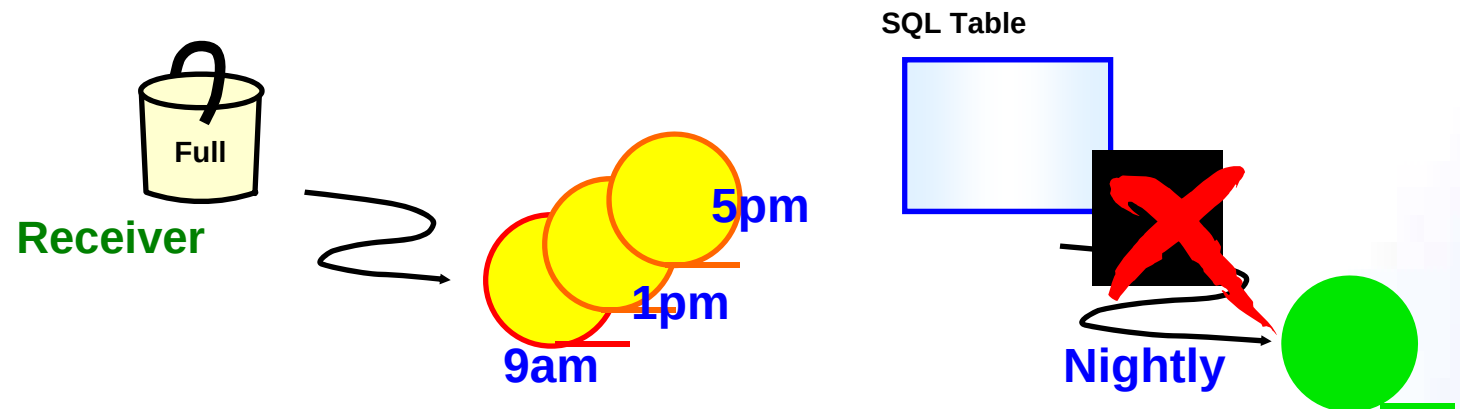


## How might this affect my Backup Procedures?

### Could save **only** journal receivers daily

- Generally smaller and fewer than PFs
- Reduces save time and media

### Would still save entire Library regularly (weekly)



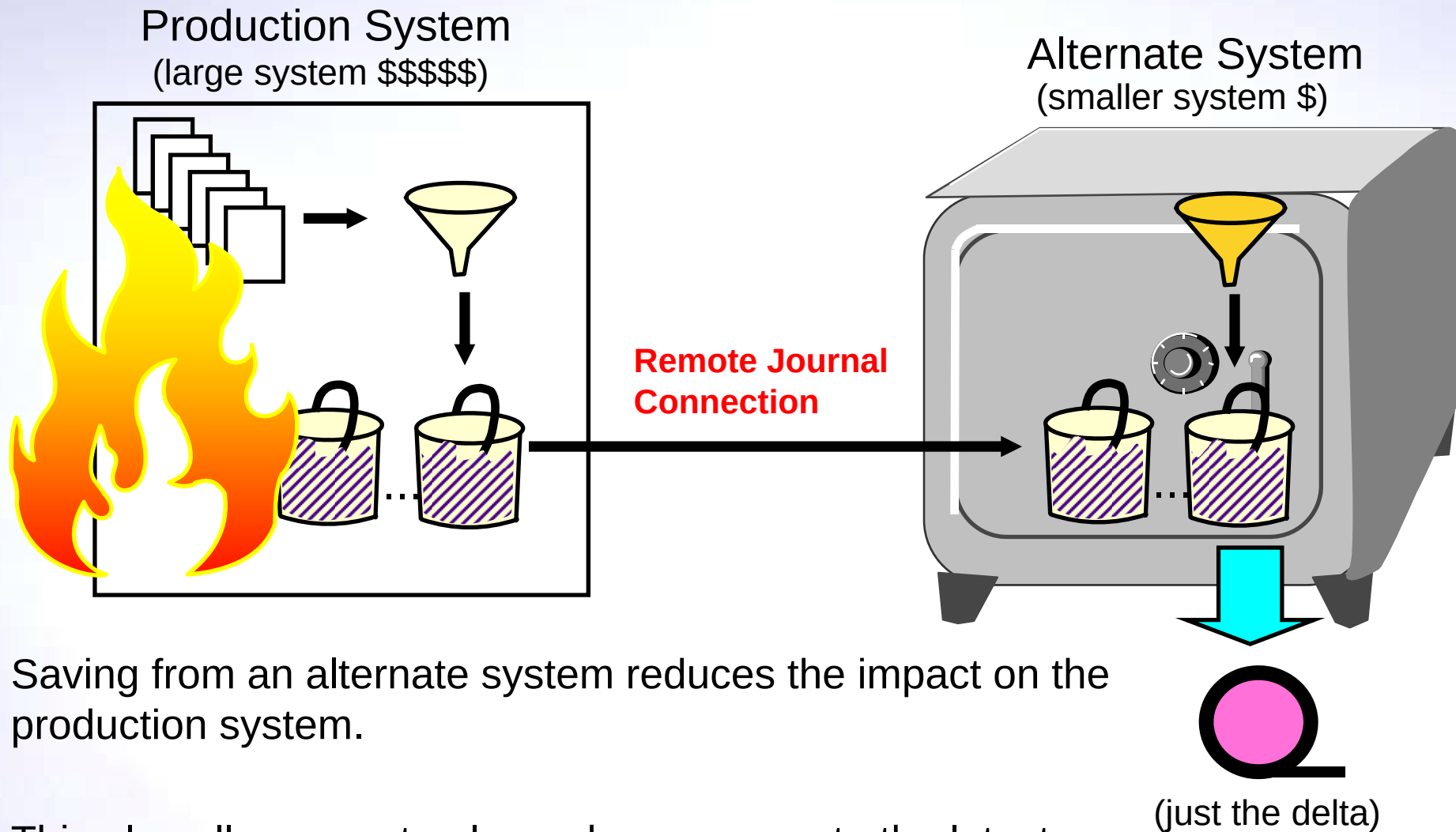
*Journal users know that there are certain housekeeping chores...*

*They know it's prudent to save off a copy of their most recent journal receiver every hour.*

*But...*

*How many really do so ?*

# Off-Site Vaulting



Saving from an alternate system reduces the impact on the production system.

This also allows you to always have access to the latest changes to apply even if you haven't saved your Journal Receivers lately!

# Disaster Recovery

- Obviously you're going to want your critical journal receivers off-site when the fire breaks out

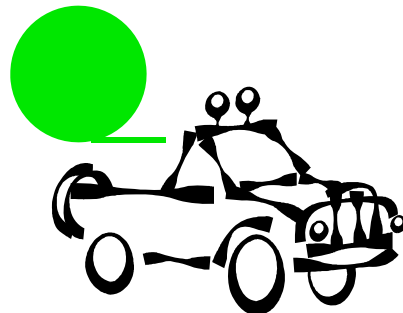
*What am I going to do after a disaster?*



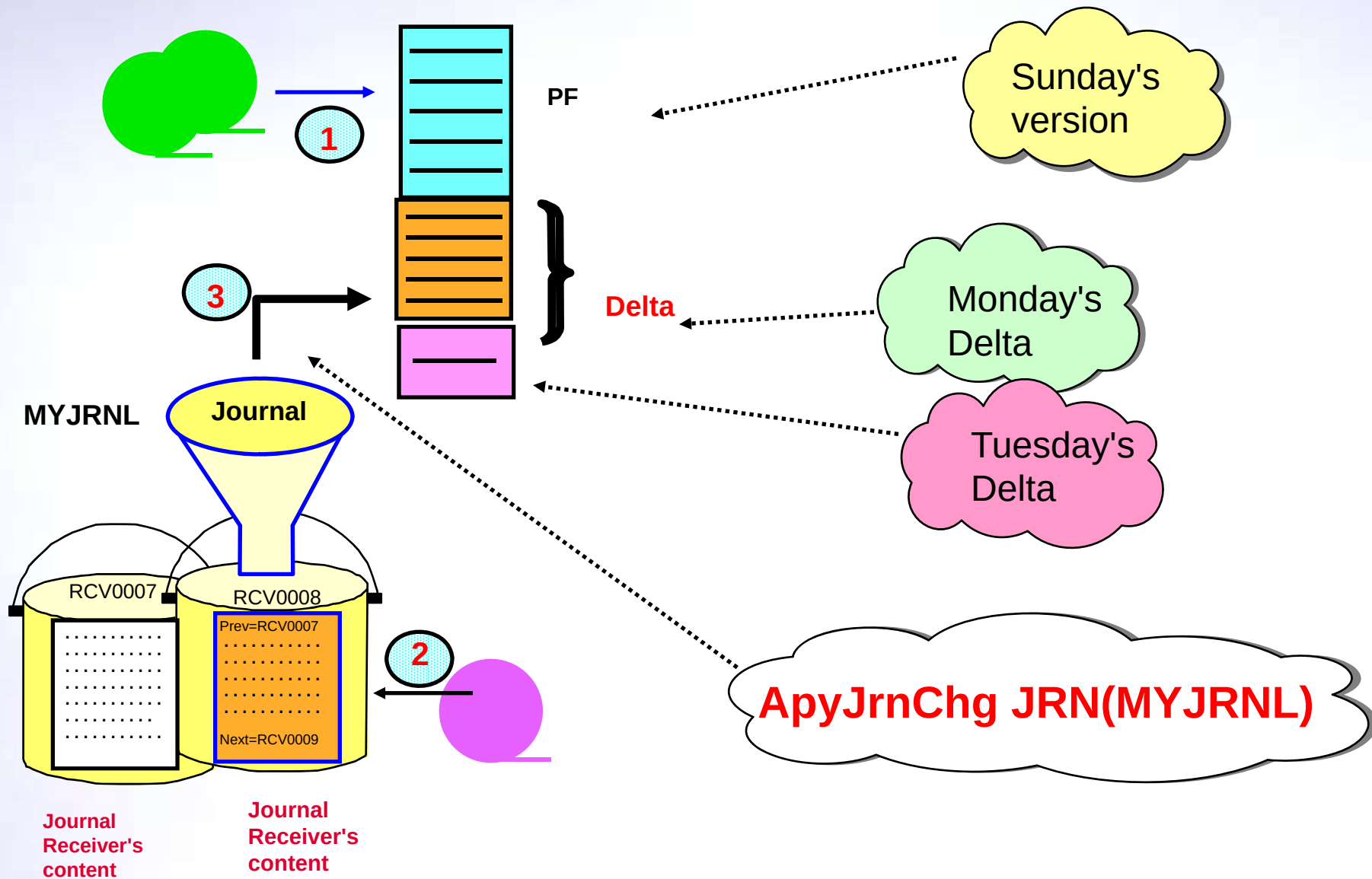
Data Center



Remote Recovery Center



## Recovery steps at the Hot site



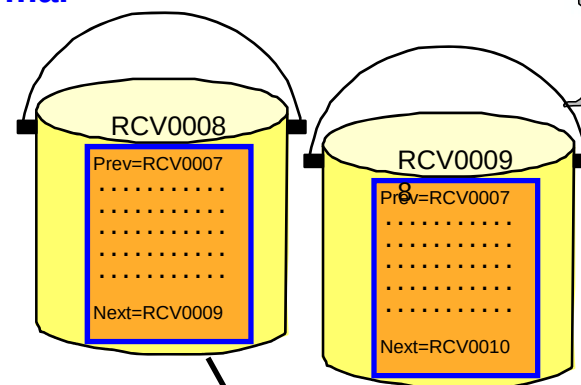
# Applying Journal Changes

## • **APYJRNCHG** command applies changes:

- Across a designated range of Receiver(s)
- From **Last save** to **most recent restore**
- By Sequence number range
- By Date/time range
- To the point you Opened or closed a file by job
- To a Commit boundary

**APYJRNCHG JRN(MYJRNL) FILE(LIBA/\*ALL)**  
**RCVRNG(RECV0008 \*CURRENT)**  
**FROMENT(\*FIRST)**  
**TOJOB(001234/USER/DSP07)**  
**CMTBDY(\*YES)**

Normal



Journal Receiver's content



PF

Delta

Applies Forward

## Can I Remove Changes ?

Maybe you've got a programmer in your shop like all-thumbs Charlie..



Yes

You'll need **BEFORE** images

{We'll talk about those in a few minutes}

How do you convince the Jrn to Remove changes made in error ?

- **RMVJRNCHG** removes changes

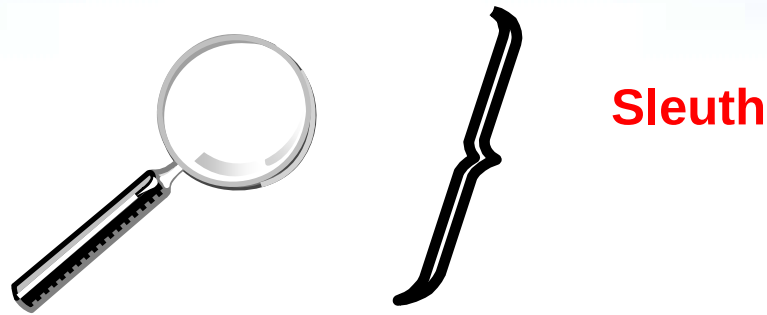


## ***Additional Uses for Journal***

# Additional benefits of journal use...

## Auditing:

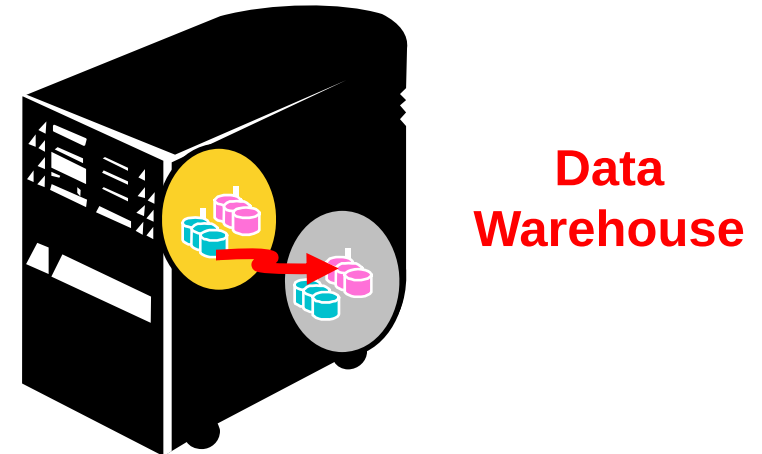
- Audit trails
- **Debugging aid**
- Security verification
  - **Intrusion Detection**



Sleuth

## Distribution across partitions:

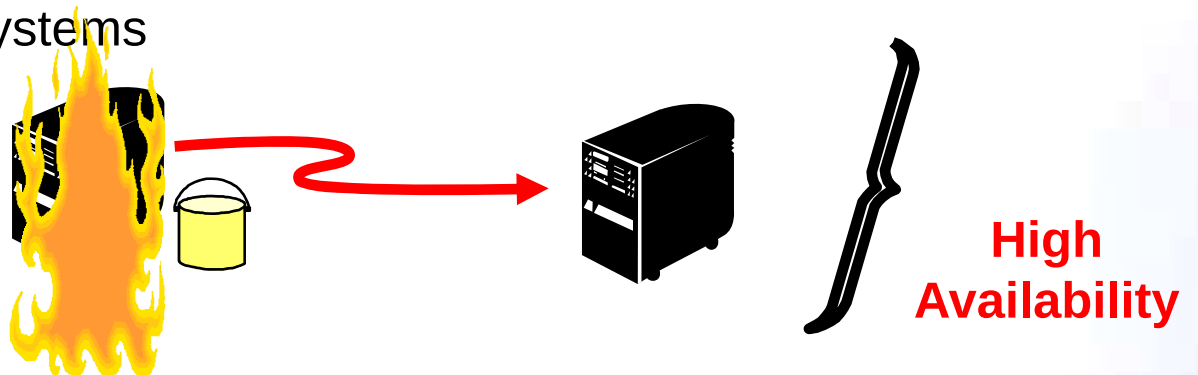
- Sent to your Data Warehouse
  - Useful for Queries and reports



Data  
Warehouse

## High Availability:

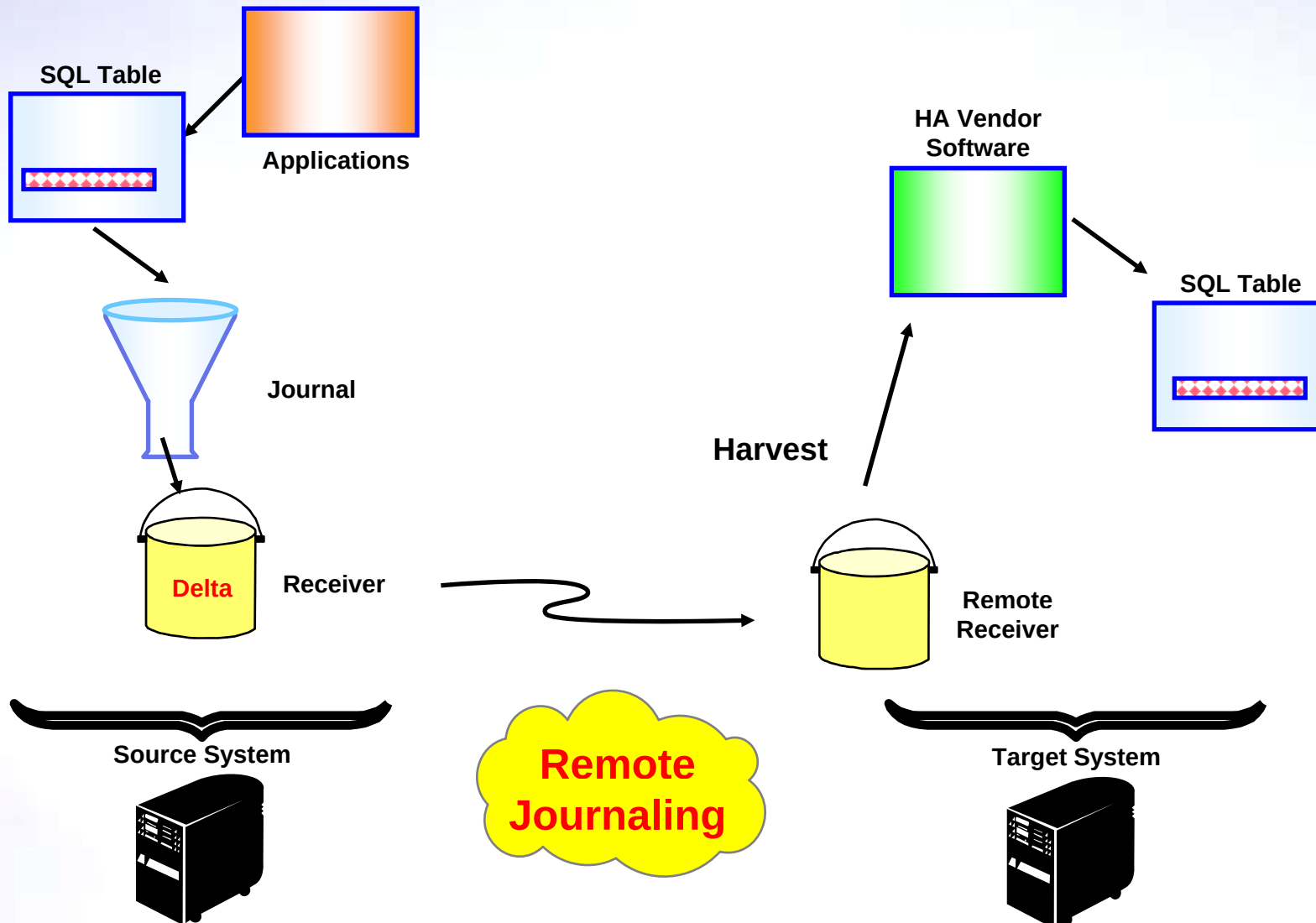
- Distribution of data to other systems
  - **24 x 7 availability**



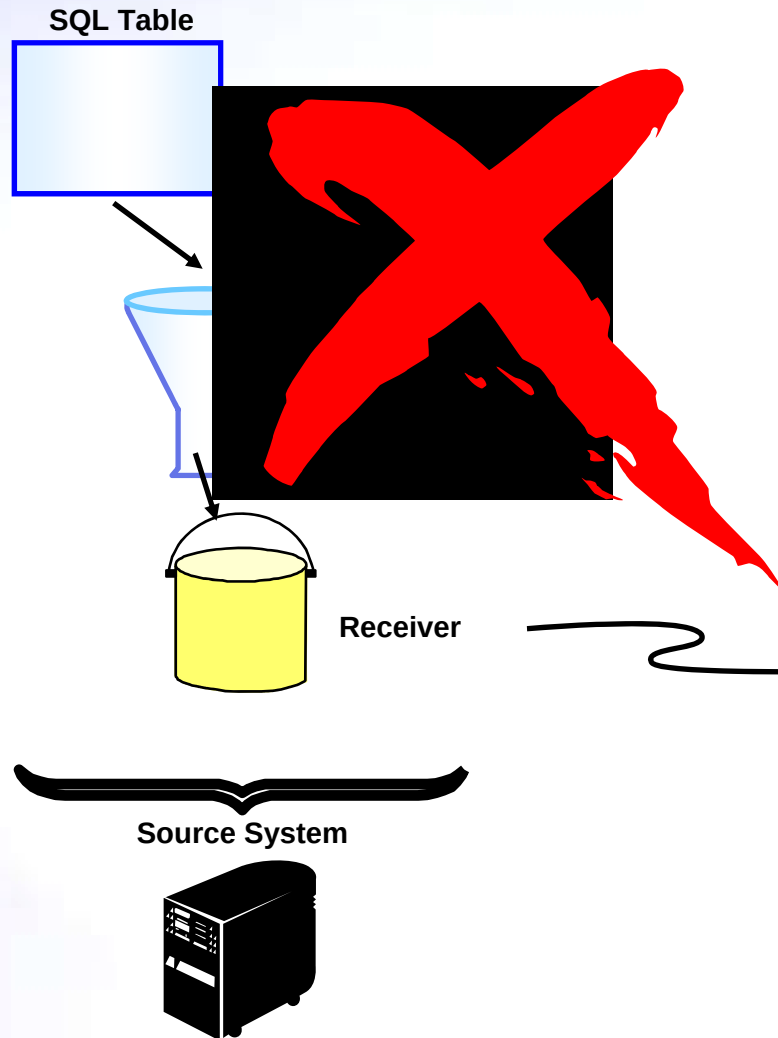
High  
Availability

**Sooner or later...**

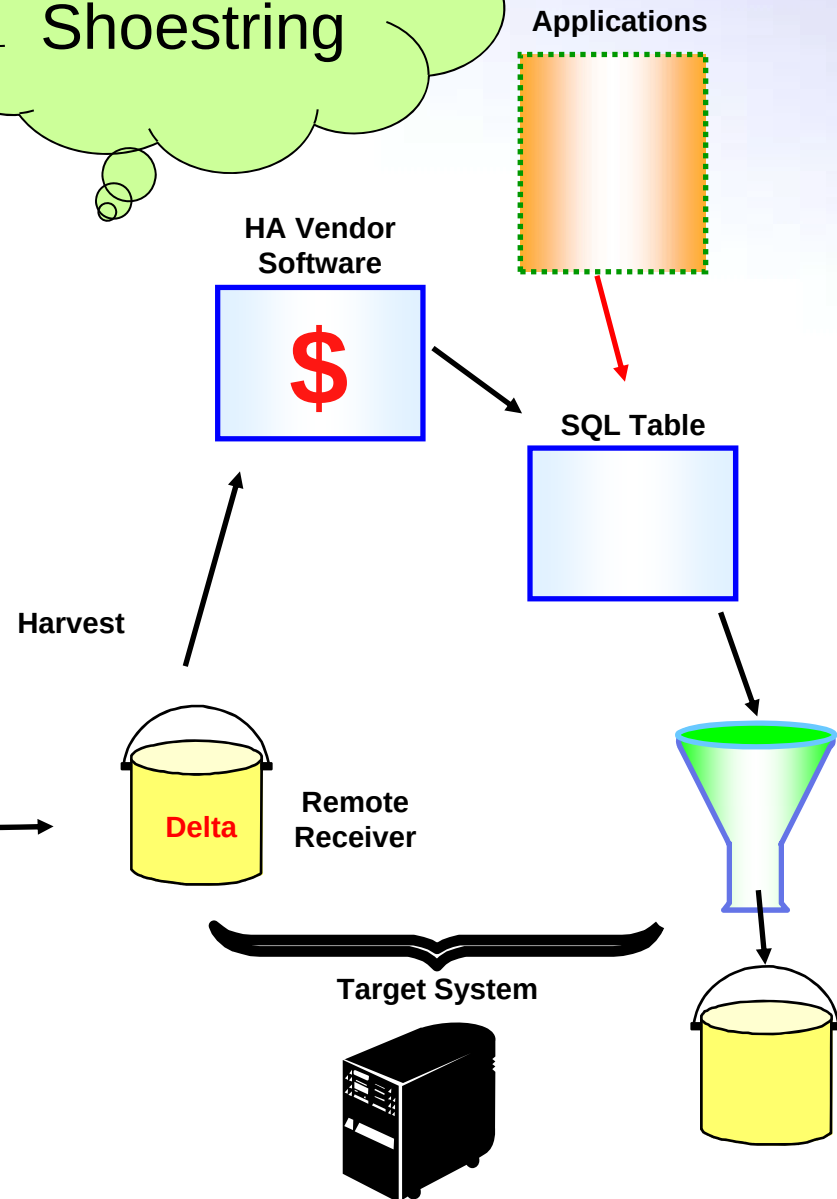
# A High Availability environment



## Role Swap



Or... Use Shoestring



*How do initiate journaling?*

1 Create journal receiver

- **CRTJRNRCV**

2 Create journal

- **CRTJRN**

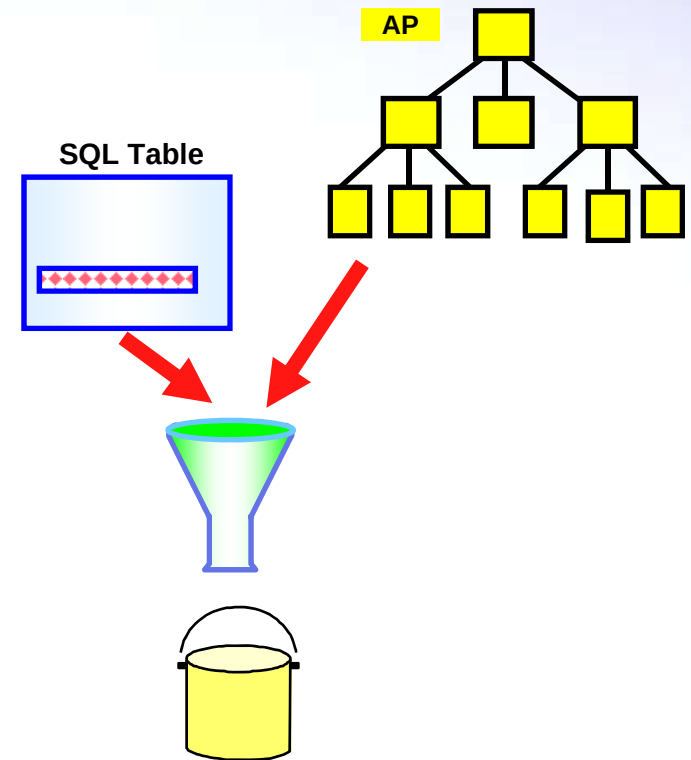
3 Identify files to be journaled

- **STRJRNPF**

4 (Optional) Identify access paths to be journaled

- **STRJRNPAP**

- ♦ Helps speed up IPL



# Enabling Journaling...

## To Start Journaling

- STRJRNPF

- STRJRNAP



**Database**

- STRJRNOBJ** (For **Data Areas** and **Data Queues**)

- STRJRN** (For IFS objects -- requires path names)

- QjoStartJournal API** (same function as STRJRN command)



**Non-DB**

How about other  
objects I want to  
protect?



## To Stop Journaling

- ENDJRNPF

- ENDJRNAP

- ENDJRN** (For IFS objects)

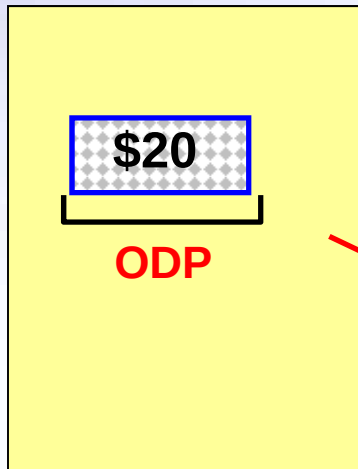
- ENDJRNOBJ** (For Data Areas and Data Queues)

- QjoEndJournal API** (same function as ENDJRN command)

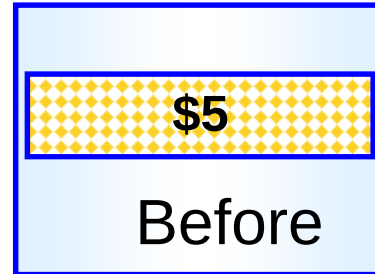
***A banking example...***

# Making a banking Deposit

Job



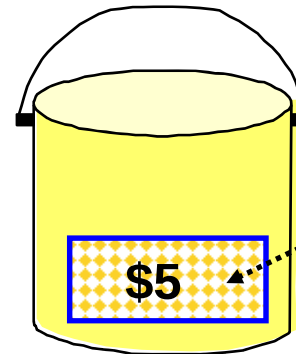
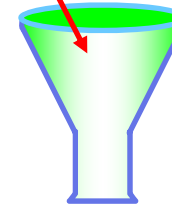
SQL Table



Choices,  
choices,  
choices...

\$20

After  
image



Before  
image

## Before and After images

Capturing both images takes more space

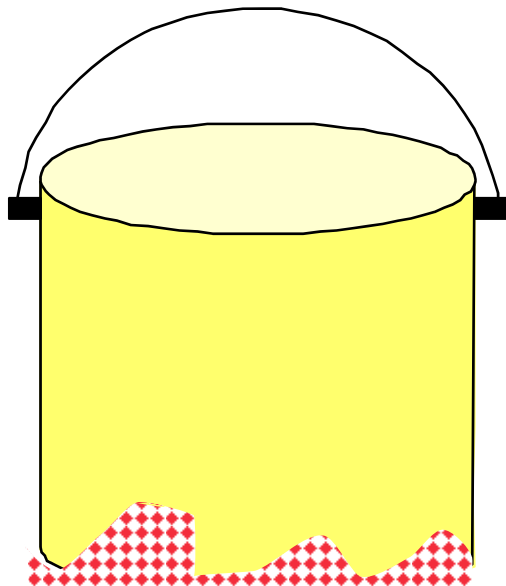
STRJRNPF . . . Images( \*AFTER )  
- Default on STRJRNPF command

STRJRNPF . . . Images( \*BOTH )  
- Default for SQL tables

Do you need both images?

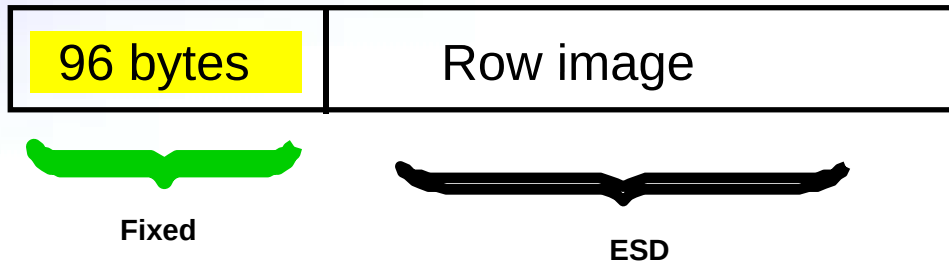


*How fast are my journal receivers going to grow?*



Receiver

# Size Considerations



There's a new  
Pseudo Journal  
tool to help you  
estimate this size

## Estimating journal receiver size

- (96 bytes + avg record length) \* # changes per day
- Twice this amount if you're capturing both before and after images

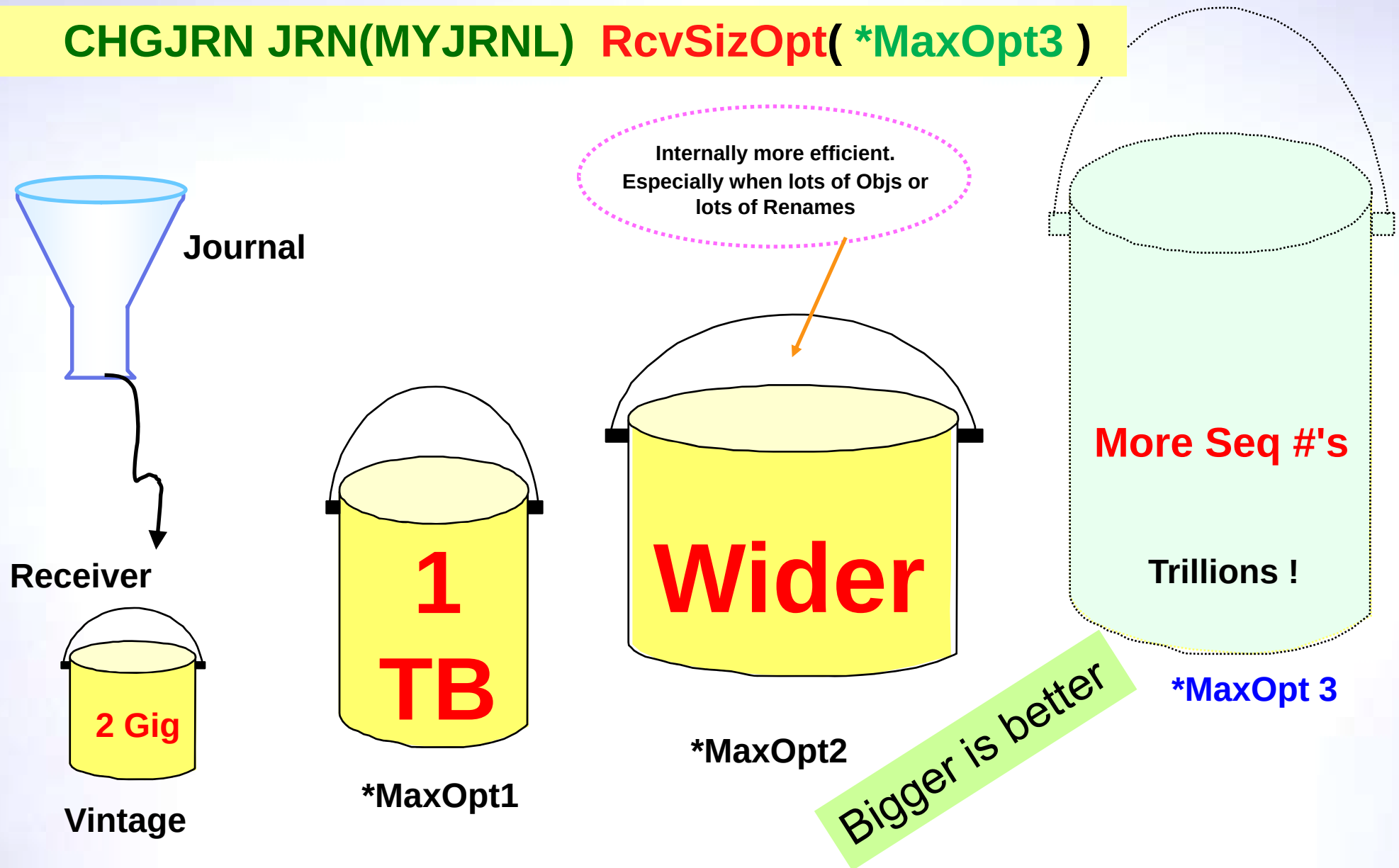


<http://www-03.ibm.com/systems/i/software/db2/journalperfutilities.html>

# Journal receivers come in a variety of sizes

## Super-size them !

**CHGJRN JRN(MYJRNL) RcvSizOpt( \*MaxOpt3 )**



*How do I manage the journal receivers?*

# Who should manage my CHGJRN ?

How often should I change receivers?

When should I change receivers?

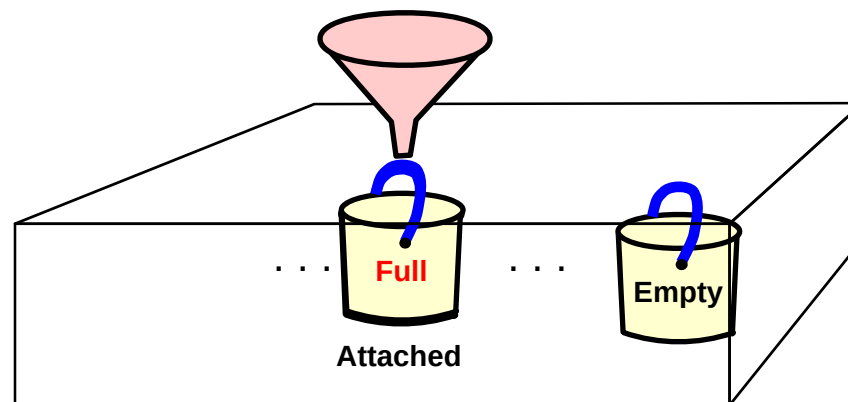
■ Ans: During a lull

What happens if I don't change receivers?

■ Consider employing **CHGJRN MNGRCV (\*SYSTEM)**



We needed  
a better  
idea



# ***The Journal Commands and Screens***

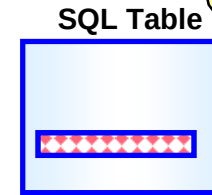
***(used to investigate your journal  
environment)***

## How Do I Get Info on Journal Environment?

Top down or bottom up?

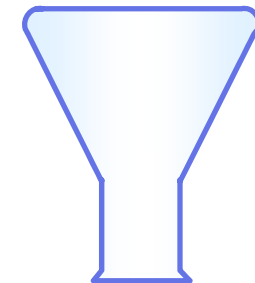
### Display File Description - **DSPFD**

- Tells you if file is journaled and journal name
  - For a data area or data queue **DSPOBJD** shows this same information
  - For an Integrated File System object **WRKLNK** shows this information



### Work with **Journal** Attributes - **WRKJRNA**

- Tells you all objects journaled through this journal
- Tells you the receiver attached to this journal

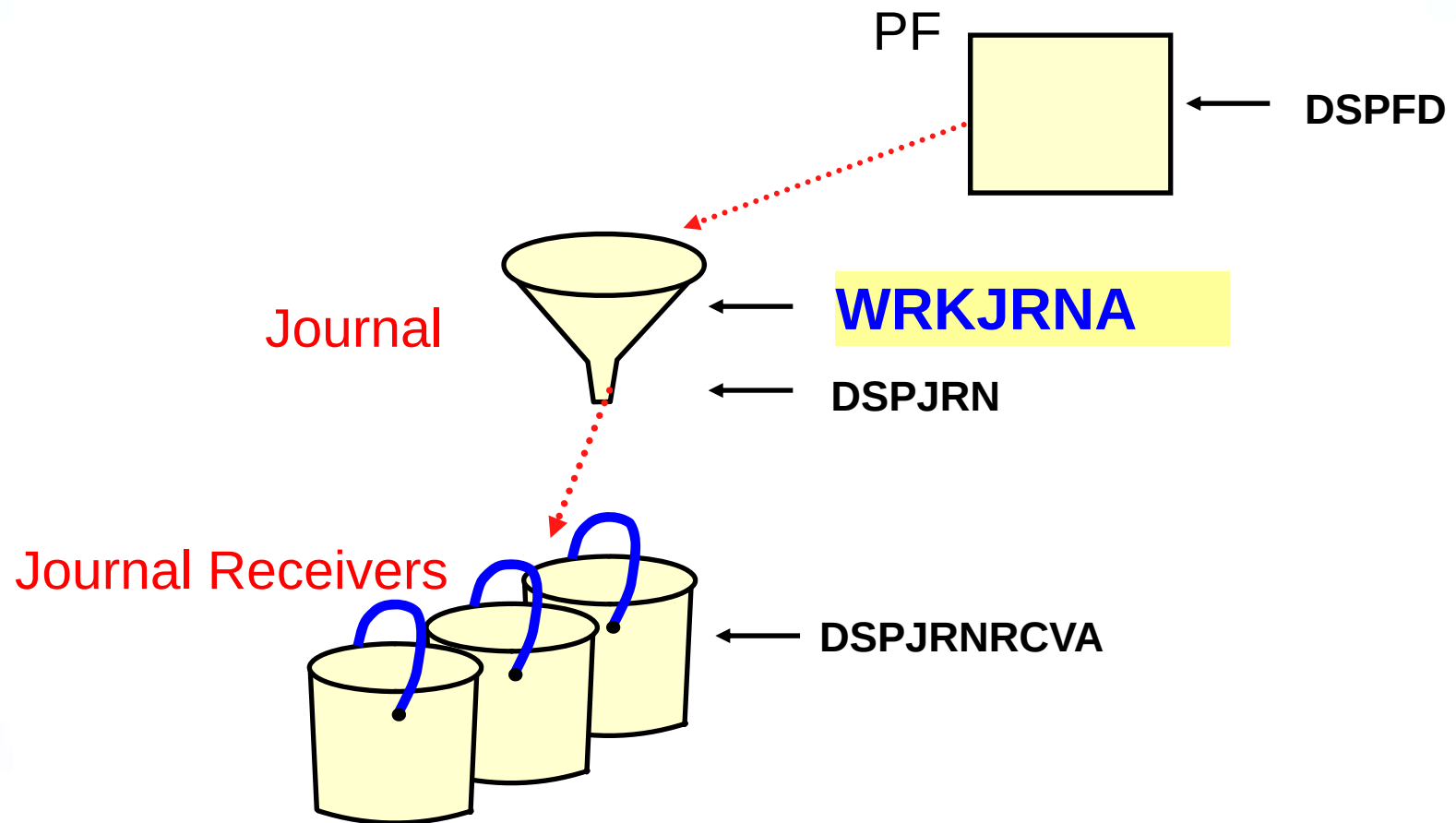


### Display Journal **Receiver** Attributes - **DSPJRNRCVA**

- Tells you to which journal the receiver is attached

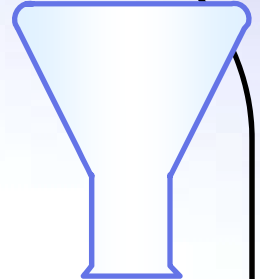


## Example of Journal Commands



# Working with journal attributes

## Work with Journal Attributes



Journal .....: TESTJRN  
Attached receiver ..: TESTRCV

Library .....: LWY  
Library .....: LWY

Text .....: \*BLANK

ASP .....: 1  
Message queue ..: QSYSOPR  
Library .....: \*LIBL  
Manage receivers ..: \*SYSTEM  
Delete receivers ...: \*NO  
Journal cache ....: \*YES

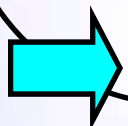
### Journalized objects:

Current .....: 10  
Maximum .....: 250000  
Recovery count ...: \*SYSDFT  
Receiver size options: \*RMVINTENT  
\*MAXOPT2

Manage delay .....: 10  
Delete delay .....: 10  
Journal type .....: \*LOCAL  
Journal state .....: \*ACTIVE  
Minimize entry data : \*NONE

Fixed length data ..: \*JOB  
\*USR  
\*PGM

**F3=Exit F5=Refresh F12=Cancel F17=Display attached receiver attributes**  
**F19=Display journaled objects F24=More keys**



## Journal Environment - Journal Attributes

Work with Journal Attributes

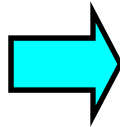
Journal .....: TESTJRN

Library .....: LWY

### Display Journalized Objects

Type choice, press Enter.

Display ..

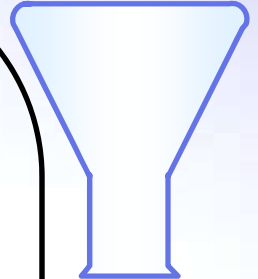


1. Files
2. Access Paths
3. Data Areas
4. Data Queues
5. Integrated File System objects
6. Commitment Definitions
7. Libraries

30. All object types

40. Summary by object type

F12=Cancel



## Journal Environment - Journal Attributes

### Display Journalled Files

Journal . . . . . : TESTJRN      Library . . . . . : LWY

Number of journalled files . . . . . : 10  
Number of journalled members . . . . . : 10

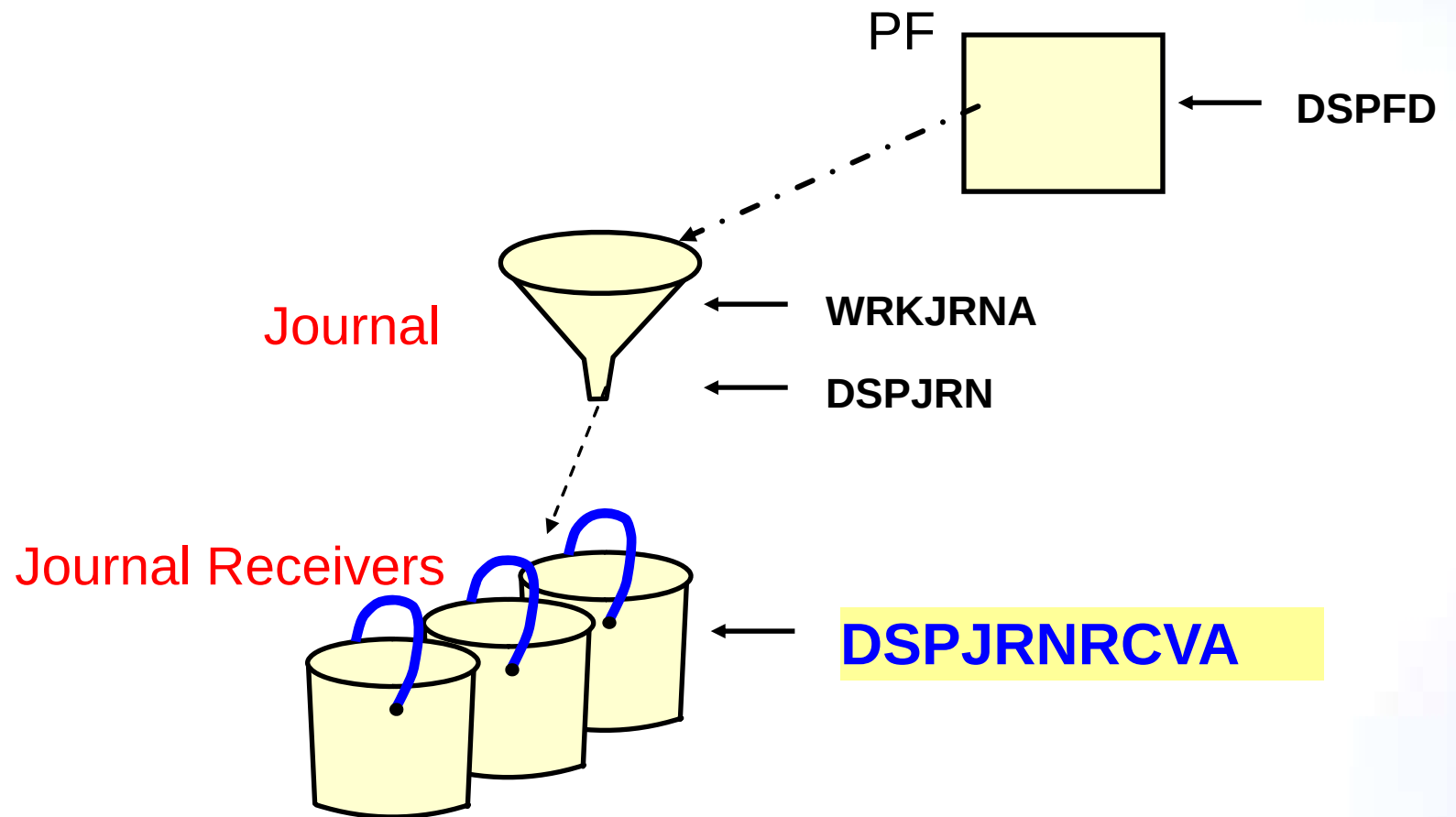
| File       | Library | File | Library | File | Library |
|------------|---------|------|---------|------|---------|
| DAVESFILE  | LWY     |      |         |      |         |
| LARRYSFILE | LWY     |      |         |      |         |
| TESTFILE1  | LWY     |      |         |      |         |
| TESTFILE2  | LWY     |      |         |      |         |
| TESTFILE3  | LWY     |      |         |      |         |
| TESTFILE4  | LWY     |      |         |      |         |
| TESTFILE5  | LWY     |      |         |      |         |
| TESTFILE6  | LWY     |      |         |      |         |
| TESTFILE7  | LWY     |      |         |      |         |
| TESTFILE8  | LWY     |      |         |      |         |

Bottom

Press Enter to continue.

F3=Exit    F5=Refresh    F12=Cancel

## What can I learn about the Receivers ?



## Journal Environment - Receiver Attributes

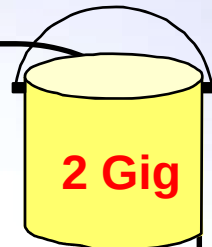
### Display Journal Receiver Attributes

|                         |          |                         |          |
|-------------------------|----------|-------------------------|----------|
| Receiver . . . . . :    | TESTRCV  | Library . . . . . :     | LWY      |
| Journal . . . . . :     | TESTJRN  | Library . . . . . :     | LWY      |
| Threshold (K) . . . . : | *NONE    | Size (K) . . . . . :    | 11268    |
| Attach date . . . . . : | 05/24/09 | Attach time . . . . . : | 17:19:28 |
| Detach date . . . . . : | 00/00/00 | Detach time . . . . . : | 00:00:00 |
| Save date . . . . . :   | 00/00/00 | Save time . . . . . :   | 00:00:00 |
| Text . . . . . :        | *BLANK   |                         |          |

|                                                |          |
|------------------------------------------------|----------|
| Auxiliary storage pool . . . . . :             | 1        |
| Status . . . . . :                             | ATTACHED |
| Number of entries . . . . . :                  | 53448    |
| Minimized fixed length . . . . . :             | NO       |
| Receiver maximums option . . . . . :           | 0        |
| Maximum entry specific data length . . . . . : | 448      |
| Maximum null value indicators . . . . . :      | 0        |
| First sequence number . . . . . :              | 62       |
| Last sequence number . . . . . :               | 53509    |

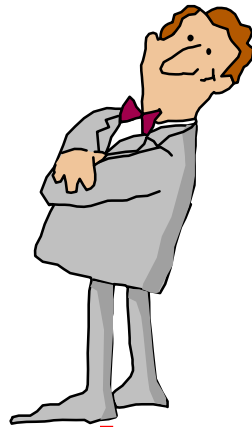
More...

F3=Exit    F5=Refresh    F6=Display associated receivers  
F10=Work with journal attributes    F12=Cancel



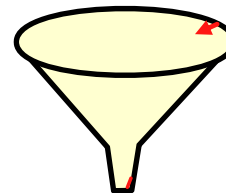
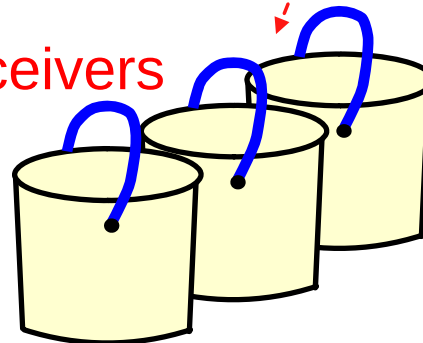
## How can I view the actual Row images ?

How can I see what flows into the Receivers?

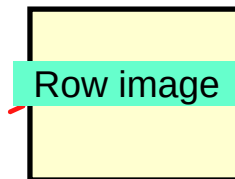


Journal Receivers

Journal



PF



DSPFD

WRKJRNA

DSPJRN

DSPJRNRCVA

# What Information is Stored?

## *Journal Entry* Contains:

| <u>Seq</u> | <u>Code</u> | <u>Type</u> | Obj   | Lib   | <u>Job</u> | <u>Time</u> | <u>Pgm</u> | Entry-specific data    |
|------------|-------------|-------------|-------|-------|------------|-------------|------------|------------------------|
| 455        | F           | OP          | FILEA | MYLIB | DSP07      | 14:36:04    | PGMA       |                        |
| 456        | R           | PT          | FILEA | MYLIB | DSP07      | 14:38:12    | PGMA       | 'This is record image' |

### Sequence Number

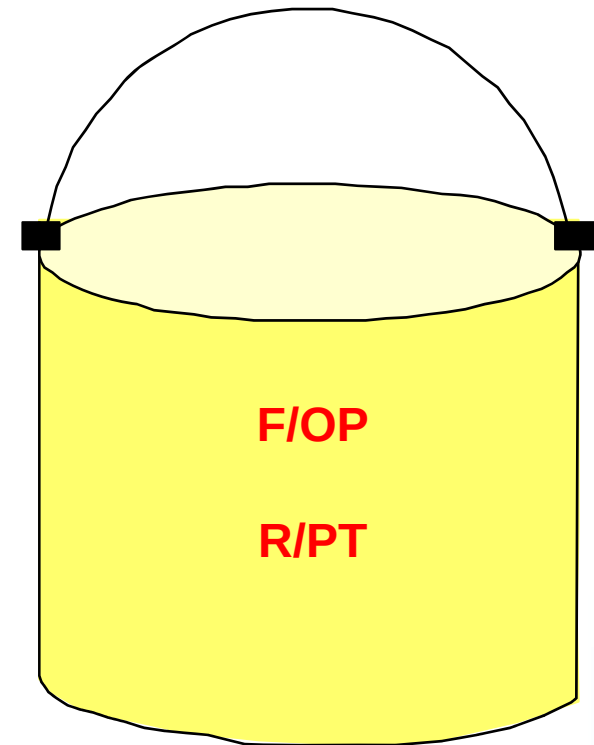
- Generated by system
- Shows order of arrival

### Journal code - 1 char, eg:

- F = File activity (open, close)
- J = Journal activity (replacing receiver)
- **R** = Record activity (add new record)

### Entry Type Code - 2 chars, eg:

- OP = Open File
- NR = Next Receiver
- **PT** = Record added to File



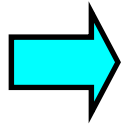
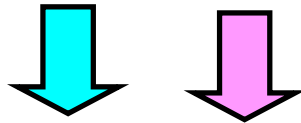
# Displaying Journal Receiver Contents

## DSPJRN

### Display Journal Entries

Journal . . . . . : TESTJRN                      Library . . . . . : LWY  
Largest sequence number on this screen . . . . . : 00000000000000000010

Type options, press Enter.  
5=Display entire entry

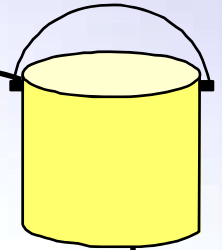


| Opt | Sequence | Code | Type | Object   | Library | Job   | Time     |
|-----|----------|------|------|----------|---------|-------|----------|
| ___ | 1        | J    | PR   | TESTFILE | LWY     | DSP07 | 14:36:04 |
| ___ | 2        | D    | JF   | TESTFILE | LWY     | DSP07 | 14:38:12 |
| ___ | 3        | F    | JM   | TESTFILE | LWY     | DSP07 | 14:38:12 |
| ___ | 5        | F    | OP   | TESTFILE | LWY     | DSP07 | 14:44:49 |
| ___ | 6        | R    | UP   | TESTFILE | LWY     | DSP07 | 14:45:06 |
| 5   | 7        | R    | UP   | TESTFILE | LWY     | DSP07 | 14:45:21 |
| ___ | 8        | R    | PT   | TESTFILE | LWY     | DSP07 | 14:45:42 |
| ___ | 9        | R    | DL   | TESTFILE | LWY     | DSP07 | 14:46:28 |
| ___ | 10       | F    | CL   | TESTFILE | LWY     | DSP07 | 14:46:31 |

F3=Exit

F12=Previous

Bottom



# Displaying Journal Receiver Contents

## Display Journal Entry

Object . . . . . : TESTFILE      Library . . . . . : LWY  
Member . . . . . : APPVEND  
Incomplete data . . : No              Minimized entry data : No  
Sequence . . . . . : 7  
Code . . . . . : R - Operation on specific record  
Type . . . . . : UP - Update, after-image

Fixed

Entry specific data  
Column    \*...+....1...+....2...+....3...+....4...+....5  
00001    'New Salary    \$100/Hr    '

ESD  
(Row Image)

Fix\_Hdr

ESD

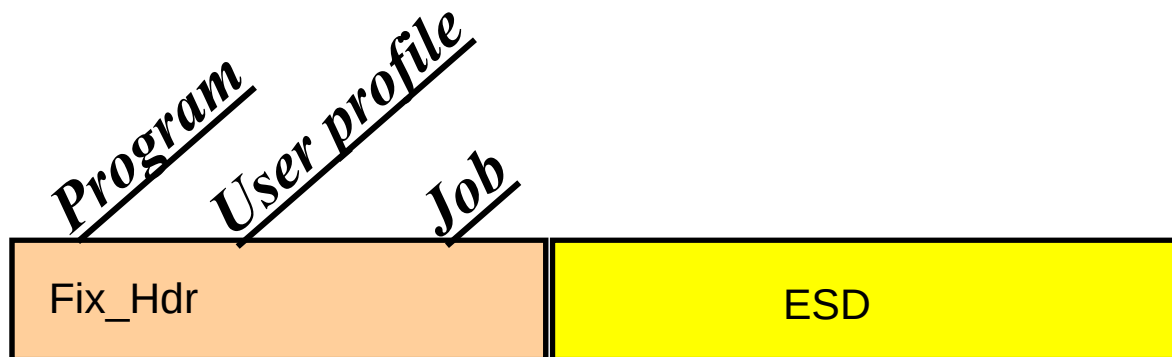
Bottom

Press Enter to continue.

F3=Exit    F6=Display only entry specific data

F10=Display only entry details    F12=Cancel    F24=More keys

# The layout of a journal entry



# Displaying Journal Receiver Contents

## Display Journal Entry

Object . . . . . : TESTFILE      Library . . . . . : LWY  
Member . . . . . : APPVEND  
Incomplete data . . : No              Minimized entry data : No  
Sequence . . . . . : 7  
Code . . . . . : R - Operation on specific record  
Type . . . . . : UP - Update, after-image

} Fixed

Entry specific data  
Column    \*...+....1...+....2...+....3...+....4...+....5  
00001    'New Salary    \$100/Hr    '

} ESD  
(Row Image)

Fix\_Hdr

ESD

Bottom

Press Enter to continue.

F3=Exit    F6=Display only entry specific data

F10=Display only entry details    F12=Cancel    F24=More keys

# Displaying Journal Receiver Contents

## Display Journal Entry Details

|                          |                                  |                          |          |
|--------------------------|----------------------------------|--------------------------|----------|
| Journal . . . . . :      | TESTJRN                          | Library . . . . . :      | LWY      |
| Sequence . . . . . :     | 7                                |                          |          |
| Code . . . . . :         | R - Operation on specific record |                          |          |
| Type . . . . . :         | UP - Update, after-image         |                          |          |
| Object . . . . . :       | TESTFILE                         | Library . . . . . :      | LWY      |
| Member . . . . . :       | APPVEND                          | Flag . . . . . :         | 0        |
| Date . . . . . :         | 9/05/09                          | Time . . . . . :         | 14:45:21 |
|                          |                                  | Ref Constraint . . . . : | No       |
| Count/RRN . . . . . :    | 5                                |                          |          |
| Job . . . . . :          | 003692/LWY/QPADEV0001            |                          |          |
| Commit cycle ID . . . :  | 0                                |                          |          |
| User profile . . . . . : | LWY                              | Program . . . . . :      | QCMD     |
| Ignore APY/RMV . . . :   | No                               | Library . . . . . :      | *OMMITED |
| Trigger . . . . . :      | No                               | ASP device . . . . . :   | *OMMITED |

} **Audit  
Info**

Press Enter to continue.

F3=Exit    F10=Display entry    F12=Cancel    F14=Display previous entry  
F15=Display only entry specific data

How fancy can I get ?

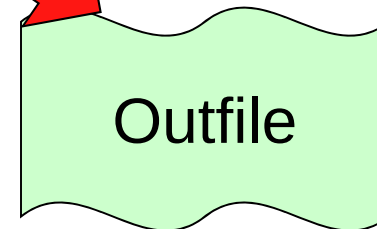
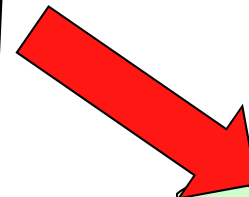
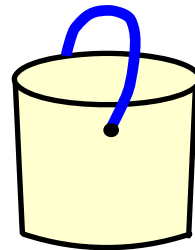
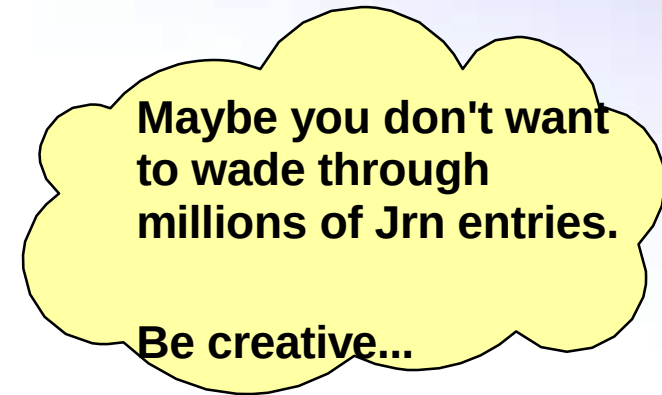
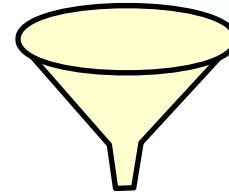
## Display Journal - **DSPJRN**

You can display journal entries by

- File
- Receiver
- Sequence number
- Journal code
- Entry type
- Job
- Program

Output to display, print, DB file

- Use to produce audit trails, activity reports



## Examples of use

- **Display changes made since 4:21 PM**

```
-DSPJRN JRN(MYJRNL) FILE(CUSTMAST)  
      FROMTIME(5/3/12' '16:21')  
      JRNCDE(R)
```

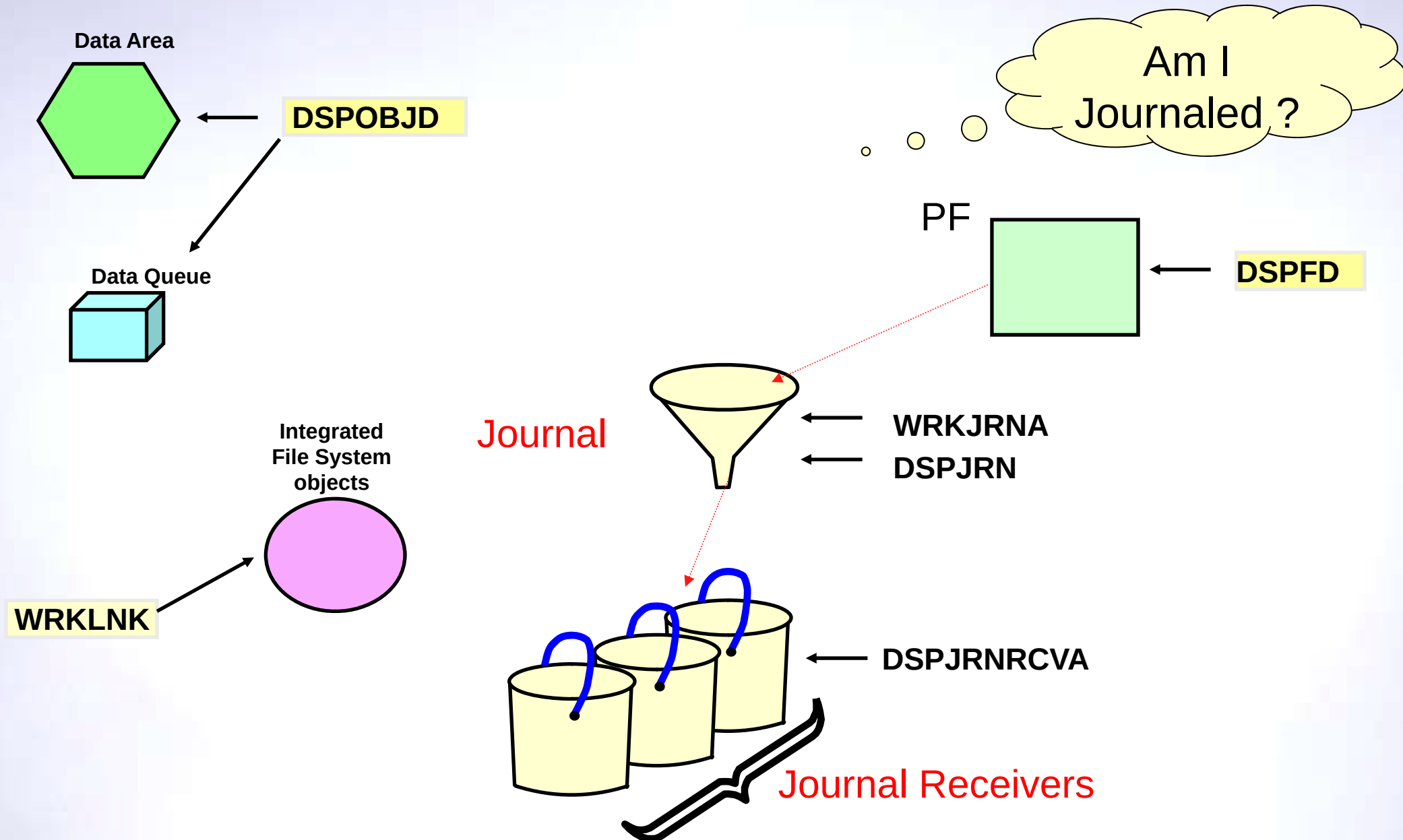
- **Display changes made by bad program**

```
-DSPJRN JRN(MYJRNL) FILE(*ALLFILE)  
      PGM(BADPGM) JRNCDE(R)
```

- **Display changes made by incomplete job**

```
-DSPJRN JRN(MYJRNL) FILE(*ALLFILE)  
      JOB(001234/USER/JOBNAME) JRNCDE(R)
```

# Other Useful Journal Commands



**What if I wanted to  
mark an event by  
generating my own  
journal entry?**

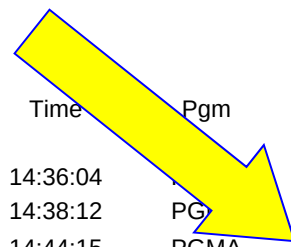


## Putting Your Own Entries Into The Journal

- You can record important application events such as:
  - Batch update completion
  - Major transactions - completion of order
- Recorded by **SNDJRNE** command
  - Produces Journal Code = "U"
- Both Entry Type and Entry Specific Data are under programmer control

```
SNDJRNE  JRN(EXAMPLE)  TYPE('CO')
          ENTDTA(&CUSTNO || &ORDNO)  FILE(CUSMAS)
```

| Seq | Code | Type | Obj    | Lib   | Job   | Time     | Pgm  | Entry-specific data    |
|-----|------|------|--------|-------|-------|----------|------|------------------------|
| 455 | F    | OP   | CUSMAS | MYLIB | DSP07 | 14:36:04 |      |                        |
| 456 | R    | PT   | CUSMAS | MYLIB | DSP07 | 14:38:12 | PG   | 'This is record image' |
| 457 | U    | CO   | CUSMAS | MYLIB | DSP07 | 14:44:15 | PGMA | 5555 ABC123'           |

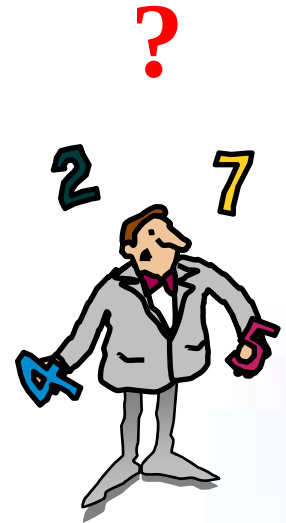
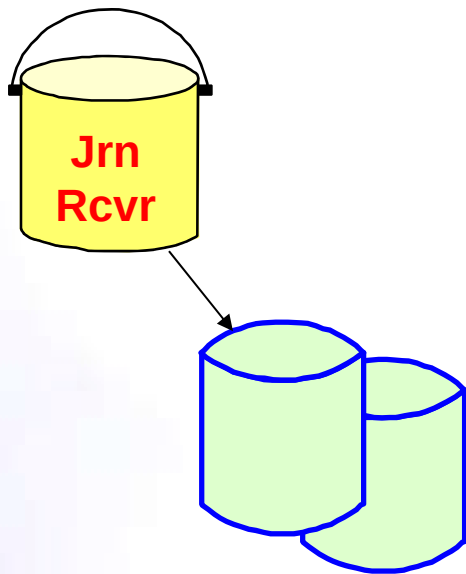


**With all this extra  
disk activity, is  
performance going  
to suffer?**



## *Can I get good performance from Journaling ?*

Yes: An overhead approaching 3% if well tuned



# The Secret to Good Journal Performance

- Two guiding principles

- **Caching**

- Don't write to disk one-by-one

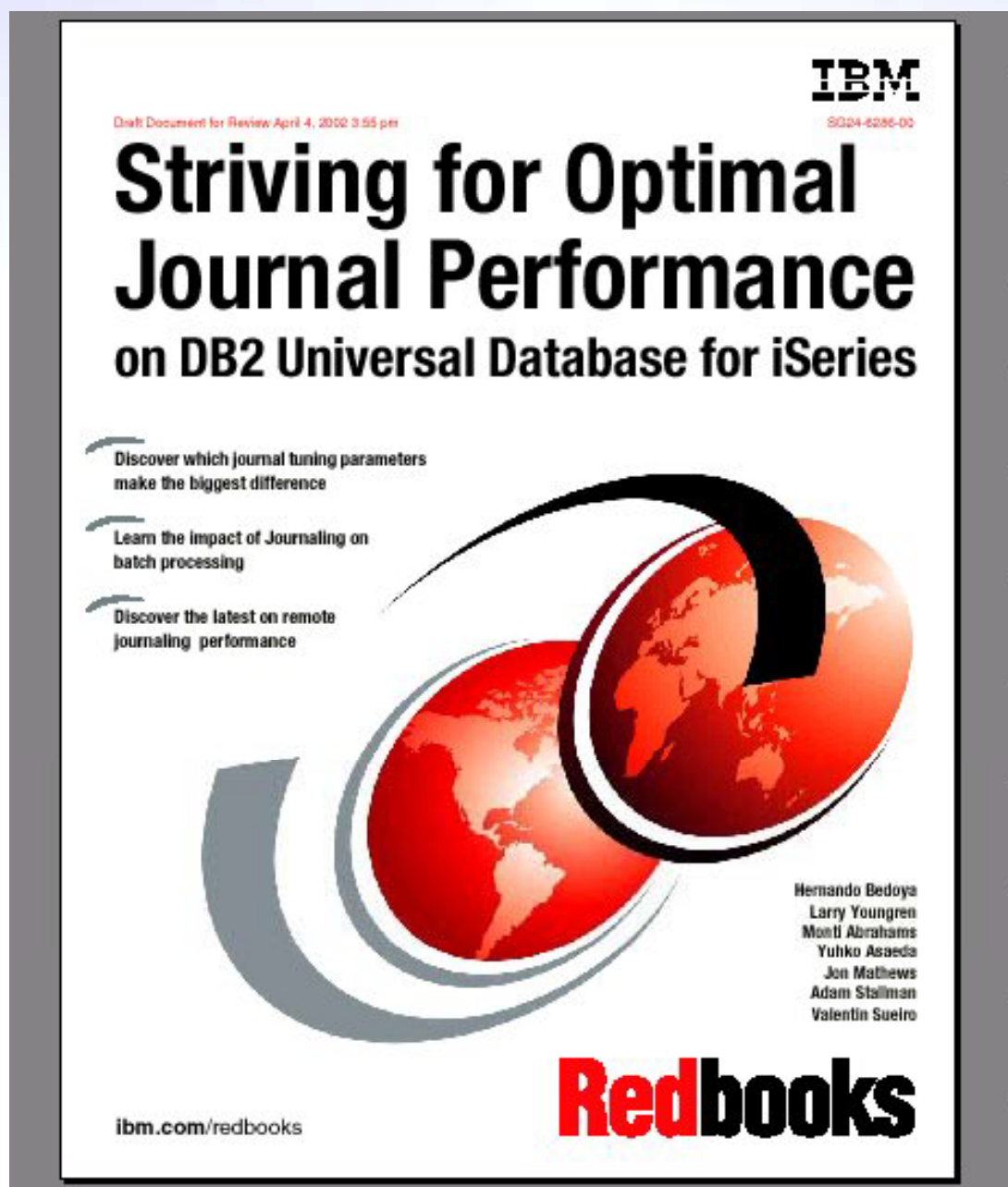
- **Parallelism**

- Spread across multiple disks

- Both your hardware configuration choices and your software tuning/overrides can make a big difference



## Primary Source of Performance hints

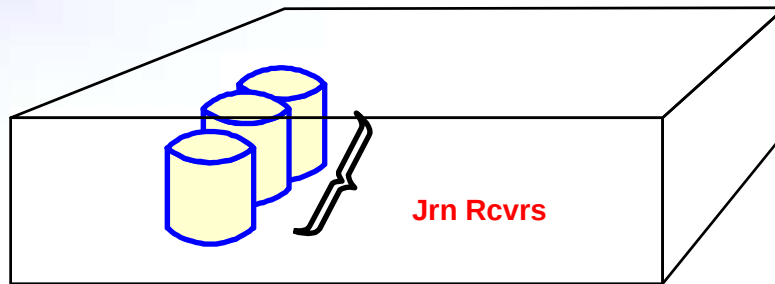


## Additional sources of Journal Performance advice

- Articles regarding **Best Journal performance practices**:
  - Have appeared in System i News
- Recent short timely journal articles on IBM's redbook site:
  - **Tech Notes**

## ***Classic Journal performance Questions***

# What About Auxiliary Storage Pools?

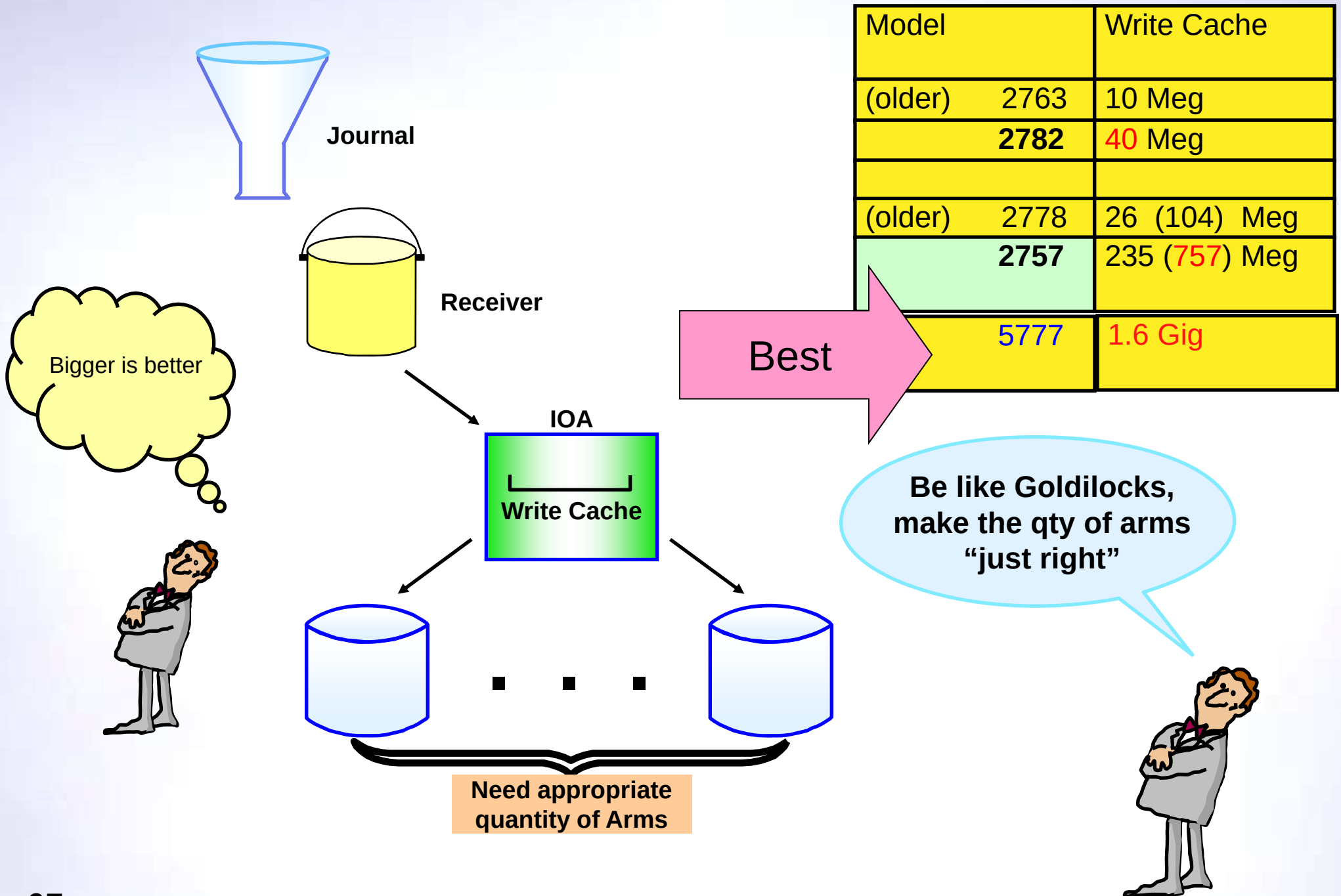


**User  
ASP**

Don't be stingy  
on disk arms



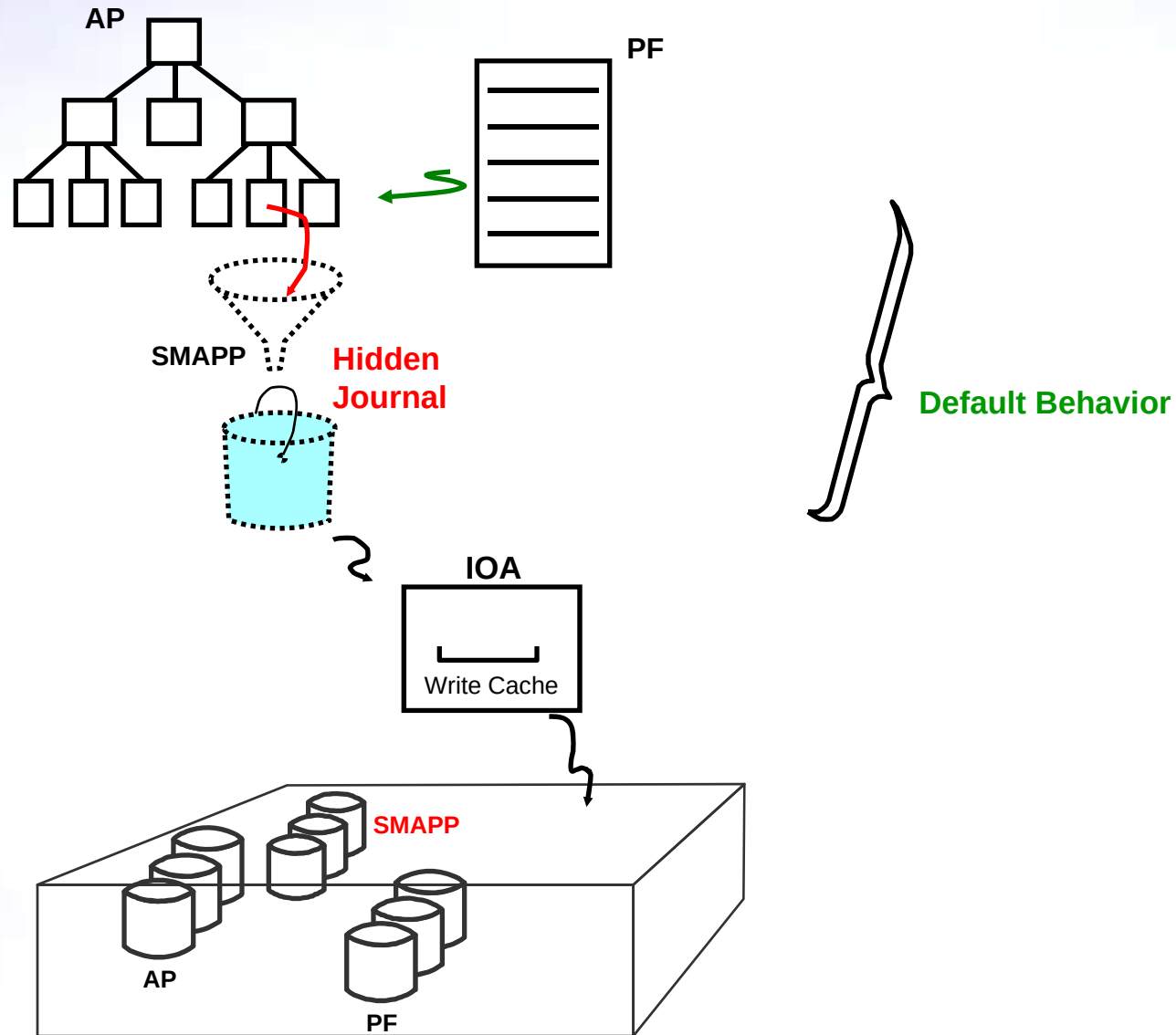
# Don't skimp on the write cache



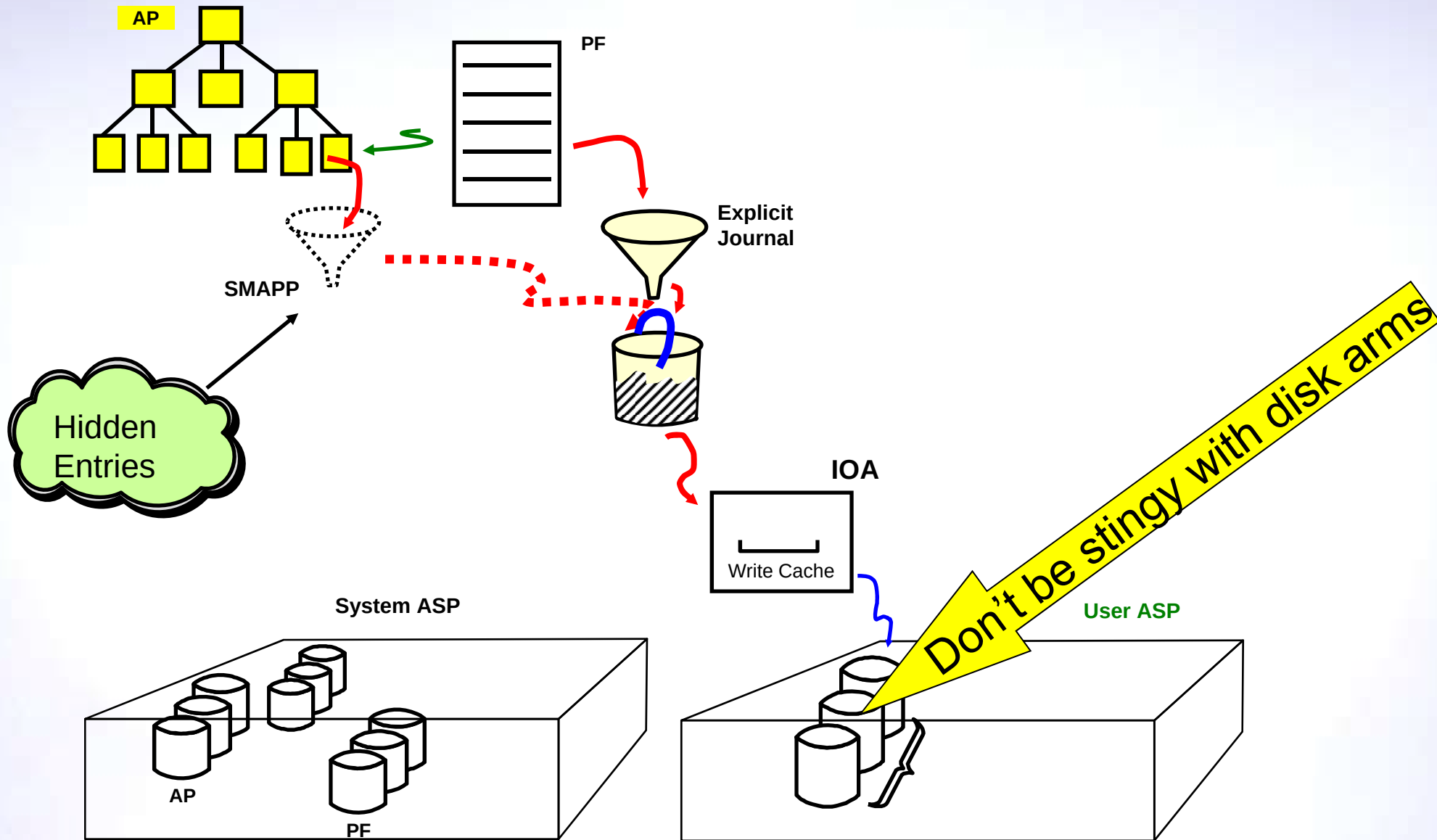
Can hidden journaling affect performance?



# System Managed Access Path Protection



## SMAPP behavior when PF is already journaled



## Why can't I see all the journal entries ?

| 1   |    | 2  |        | 4  |        | Jrn Receiver |     |
|-----|----|----|--------|----|--------|--------------|-----|
| ... | PT | PT | Hidden | DL | Hidden | UP           | ... |

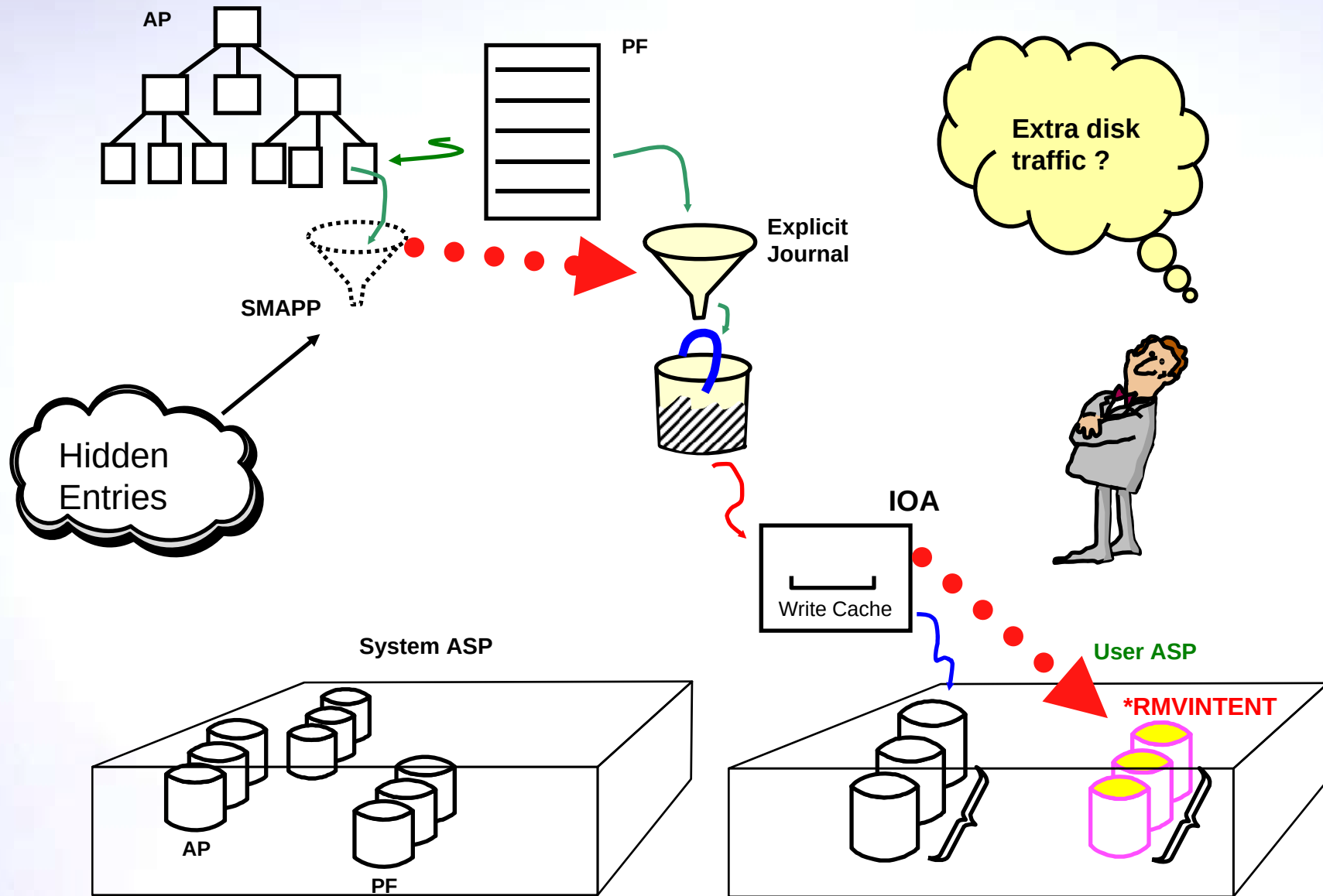
- Internal system-recovery flavored journal entries are **not** displayed
- These include:
  - **SMAPP** entries for recovery of Access paths
  - **\*REUSEDLT** entries for recalibration and recovery of deleted record counts

(Use the option to "see" these formerly hidden entries)

**DSPJRN ... IncHidEnt(\*YES)**

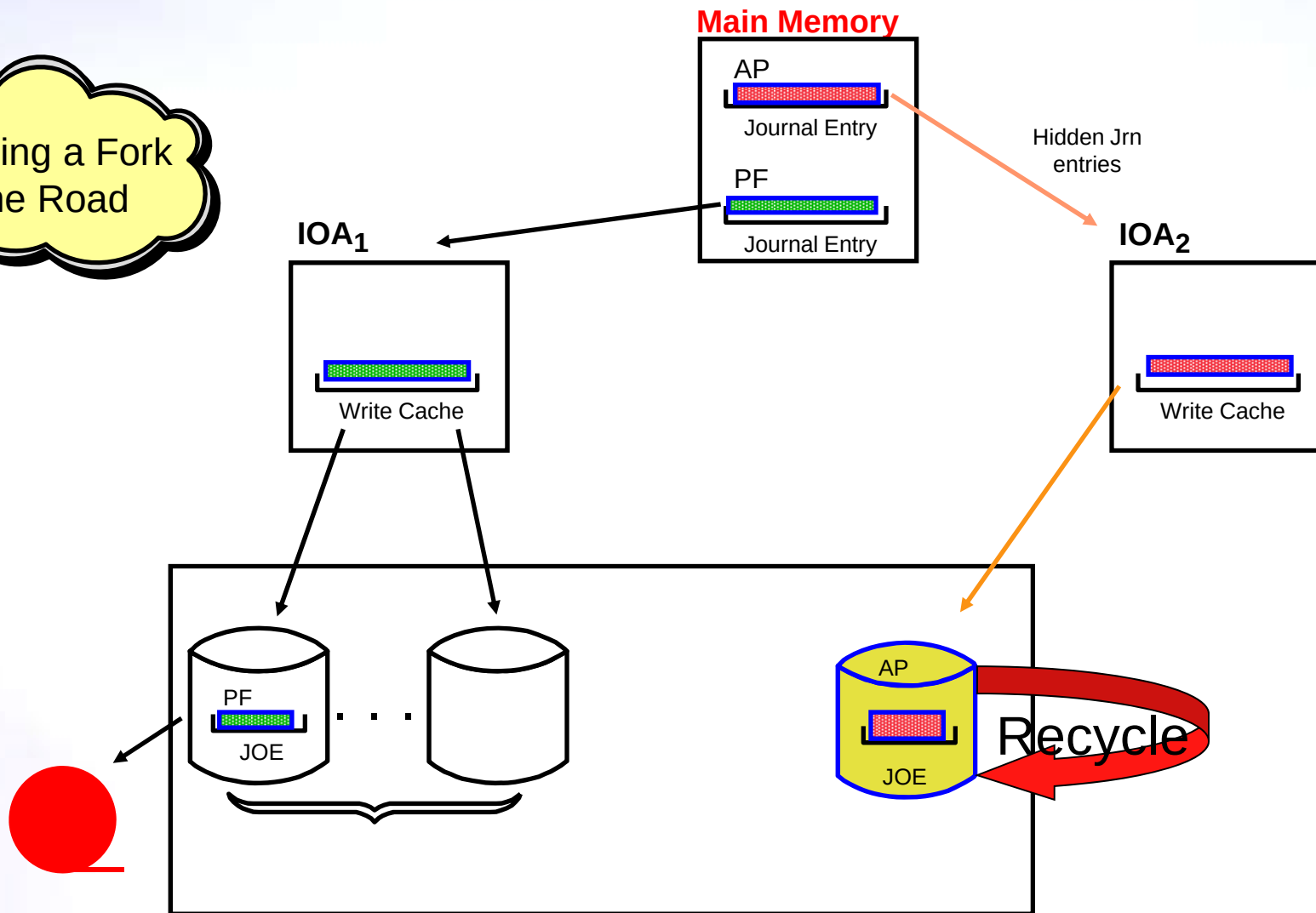
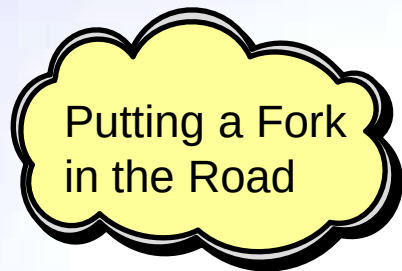
System i News article

## Throttling this hidden journaling



# Overlapped writes with

**CHGJRN ... RcvSizOpt(\*RmvIntEnt)**



*The secret turbo-charger...*

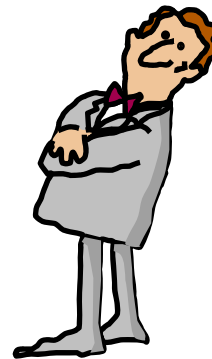
## Putting the pedal to the metal...

### Some representative Journal Deposit Rates:

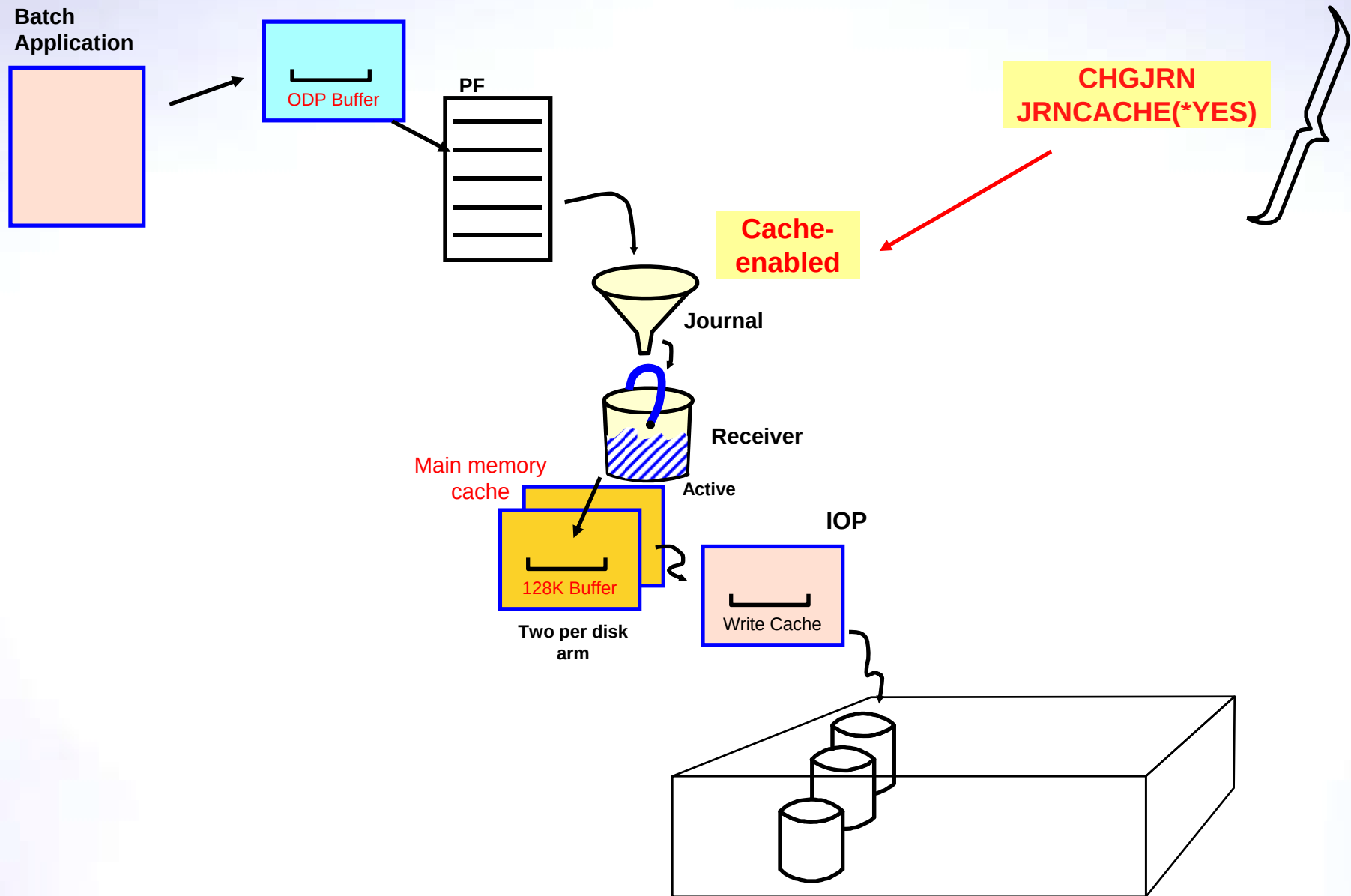
- Lab: 650 Meg/Min = 2.8 Million Journal Entries/Min
- Field: 350 Meg/Min = 1.4 Gig Every Four Minutes
- Remote Journal: **2.1 Million journal entries/minute**
  - 1 ***Billion*** Journal Entries per day

The secret is Caching

How do they achieve  
and sustain such rates  
?



# Journal Caching



## Journal caching

**Primarily aimed at reducing journal overhead for long-running batch jobs**

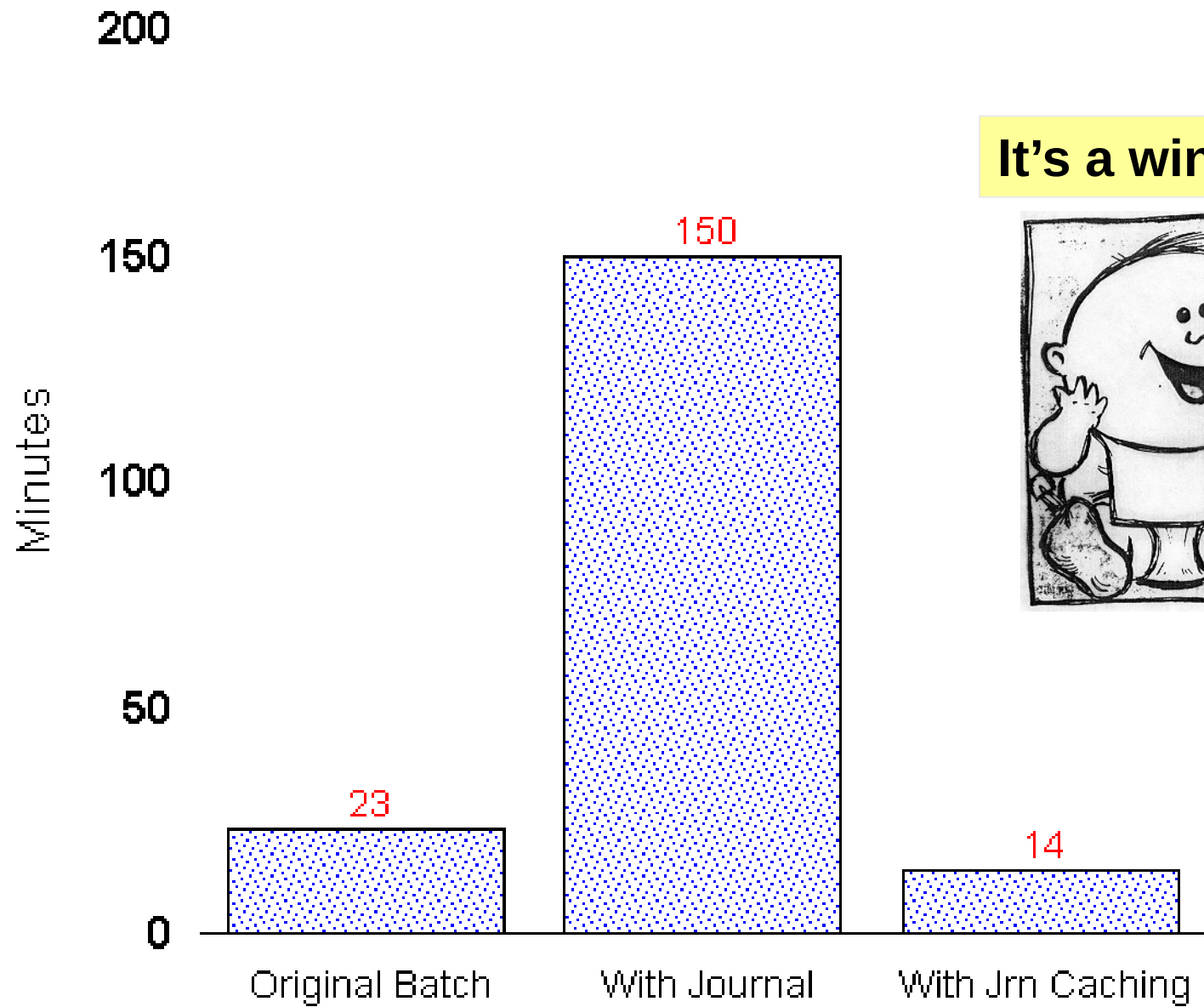
- Increases main memory caching efficiency and effectiveness
- Journal entries linger in main memory buffer until optimal size is achieved (128K)
- Corresponding database records also linger in main memory
- Caches not only within a single batch job, also caches across multiple jobs

**Buffers "Adds", "Updates", "Deletes"**

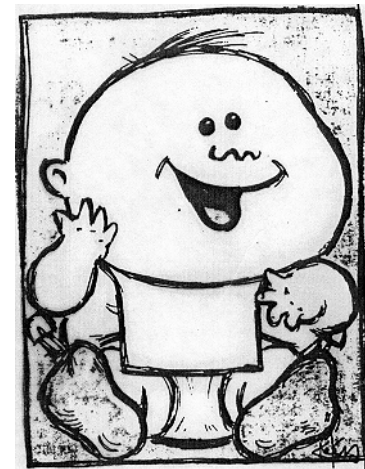
**Support is available as:**

**HA Journaling IBM i priced feature 42**

## Tuning Batch Jobs



**It's a winner**





# Introduction to Journaling

Chad Olstad

Larry Youngren

Journal\_Guru@Yahoo.com