

# ■ ■ ■ Transaction Management Internals



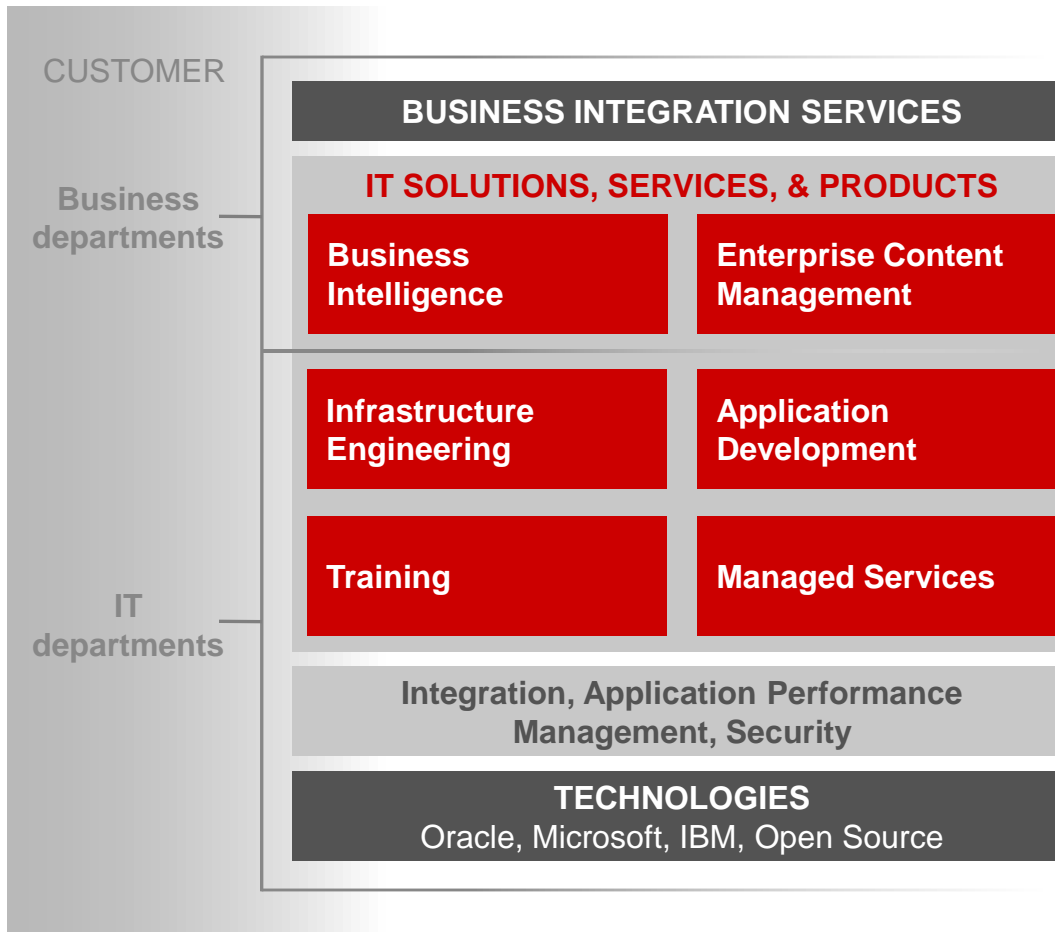
**DOAG Konferenz 2010**

Christian Antognini  
Principal Consultant

November 18, 2010  
Nürnberg (DE)

**trivadis**  
makes IT easier. ■ ■ ■

# Trivadis Facts & Figures



- 11 Trivadis locations with more than 540 employees
- Financially independent and sustainably profitable
- Key figures 2009
  - Revenue CHF 100 / EUR 66 mio.
  - Services for more than 650 clients in over 1'600 projects
  - Over 160 Service Level Agreements
  - More than 5'000 training participants
  - Research and development budget: CHF 5.0 / EUR 3.3 mio.

# Why We Are Special



---

## **Customer-specific solution competence and vendor independence**

- offers substantiated techniques and skills as well as self-developed approaches
- guarantees repeatable quality and a safe execution

---

## **Technology competence**

- offers more than 15 years of expertise in Oracle, Microsoft, IBM, and open source
- has its own Technology Center and strives for technological excellence

---

## **Solution and integration expertise**

- has a wide and cross-sectoral customer basis and more than 1600 projects every year
- Combines technological expertise with an understanding of the specific business needs of the client

---

## **Support for the entire IT project lifecycle**

- has a modular portfolio of services for the entire IT project lifecycle
- provides the appropriate combination of solutions and services for every „level of maturity“

# Who Am I



Principal consultant, trainer and partner  
at Trivadis in Zürich (CH)

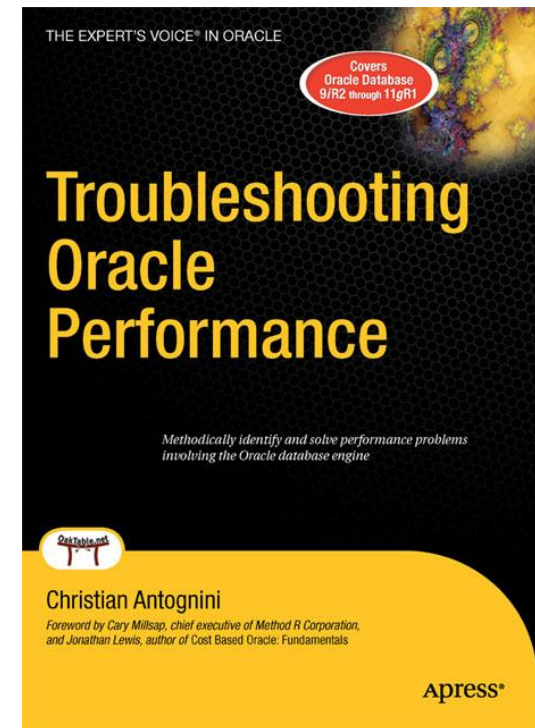
- christian.antognini@trivadis.com
- http://antognini.ch

Focus: get the most out of Oracle

- Logical and physical database design
- Query optimizer
- Application performance management and tuning
- Integration of databases with Java applications


Proud member of

- Trivadis Performance Team
- OakTable Network



# Agenda

- Concepts
- Transaction Layer Structures
- Undo Structures
- ITL Waits and Deadlocks

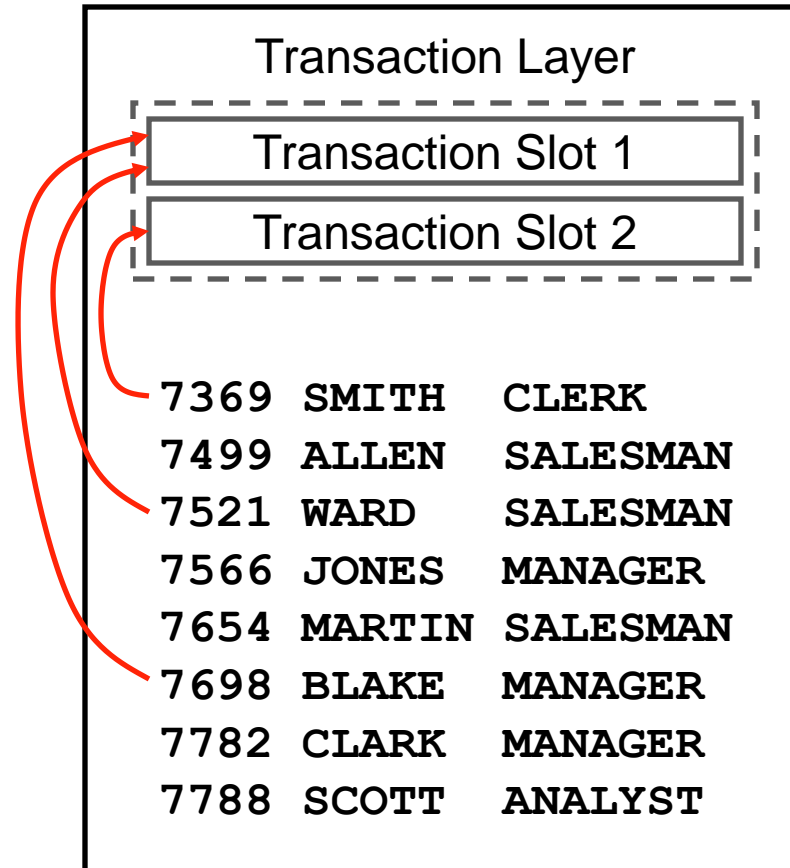


Data is always  
part of the game.

# Row Locking



## Data Block



# INITRANS and MAXTRANS



The minimum and maximum number of slots in the transaction layer depends on the INITRANS and MAXTRANS parameters.

- If free space is available, the number of slots can grow dynamically.
- It is usually better to specify a 1-2 percent higher PCTFREE than a high INITRANS!

As of 9i at least 2 slots are always created.

- The data dictionary lies to us on this matter.

The default for MAXTRANS is 255.

- As of 10g MAXTRANS is deprecated.
- The actual maximum depends on the block size (e.g. 169 with 8KB).

# Block Cleanout



When a transaction commits, some structures in the modified data blocks should be modified.

- The row lock should be cleared.
- The transaction slot should be modified to register the commit.

This operation is known as *block cleanout*.

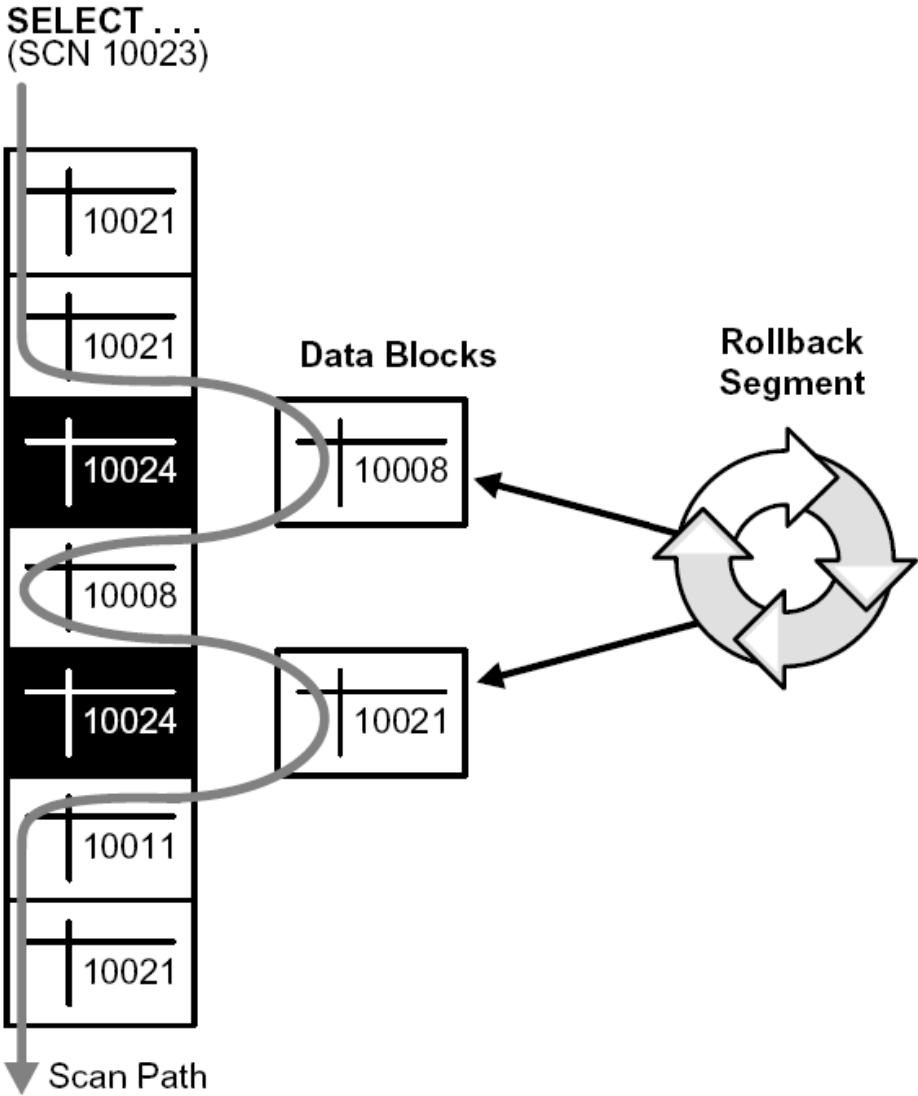
Two different approaches exist to perform the block cleanout.

- Fast: the cleanout is performed at commit time.
- Deferred: the transaction is marked as committed in the undo header only; the actual cleanout is performed the next time the block is accessed.

To perform a fast block cleanout a list referencing all modified blocks has to be managed.



# Consistent Reads



Source: Oracle Database Concepts 10g Release 1

# Physical or Logical Undo?

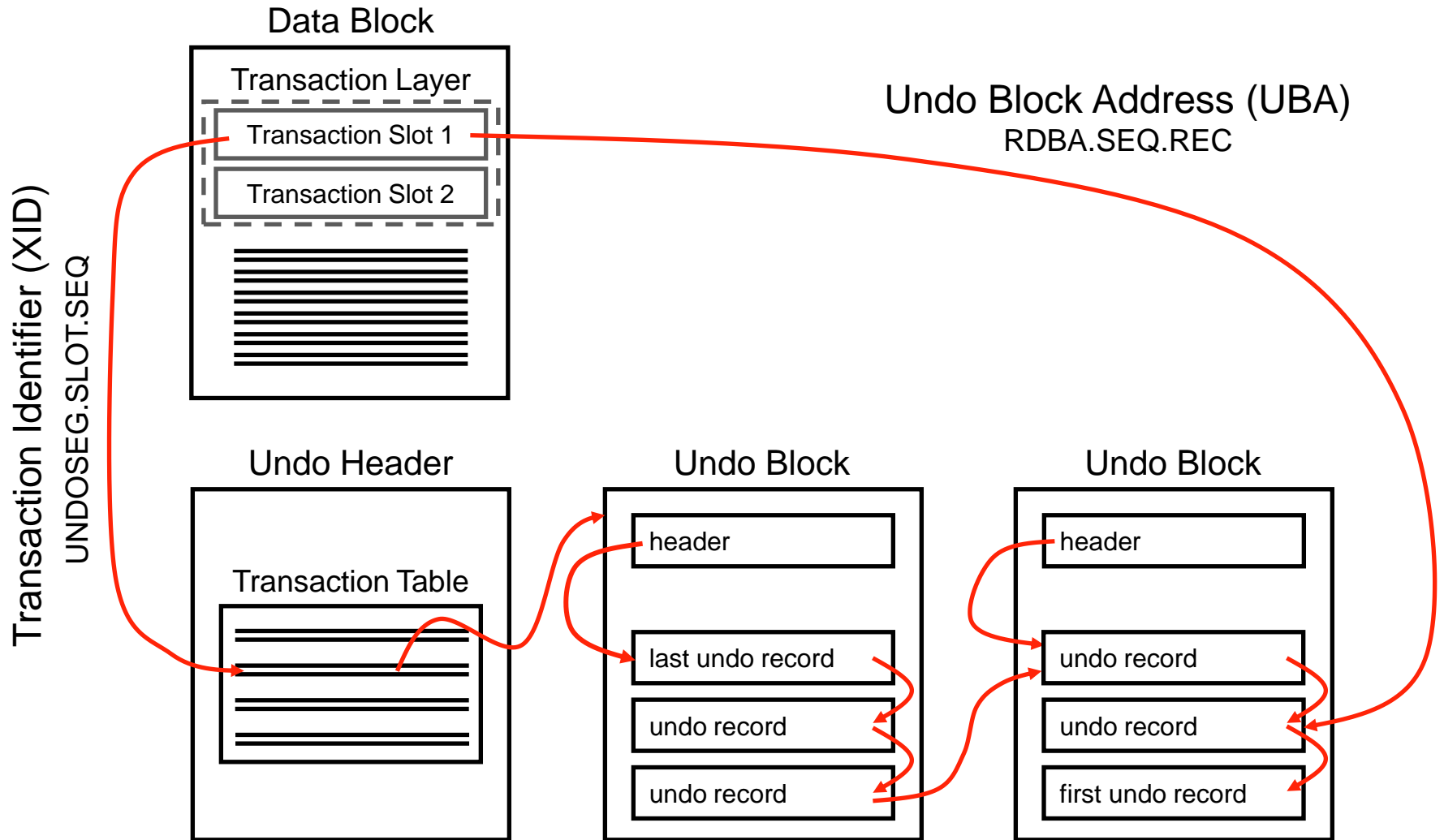


Undo segments store information about modifications and not before images of the modified blocks.

Even though a transaction is rolled back, not all modifications caused by the transaction are restored.

- E.g. an increase of the high-water mark is not rolled back.

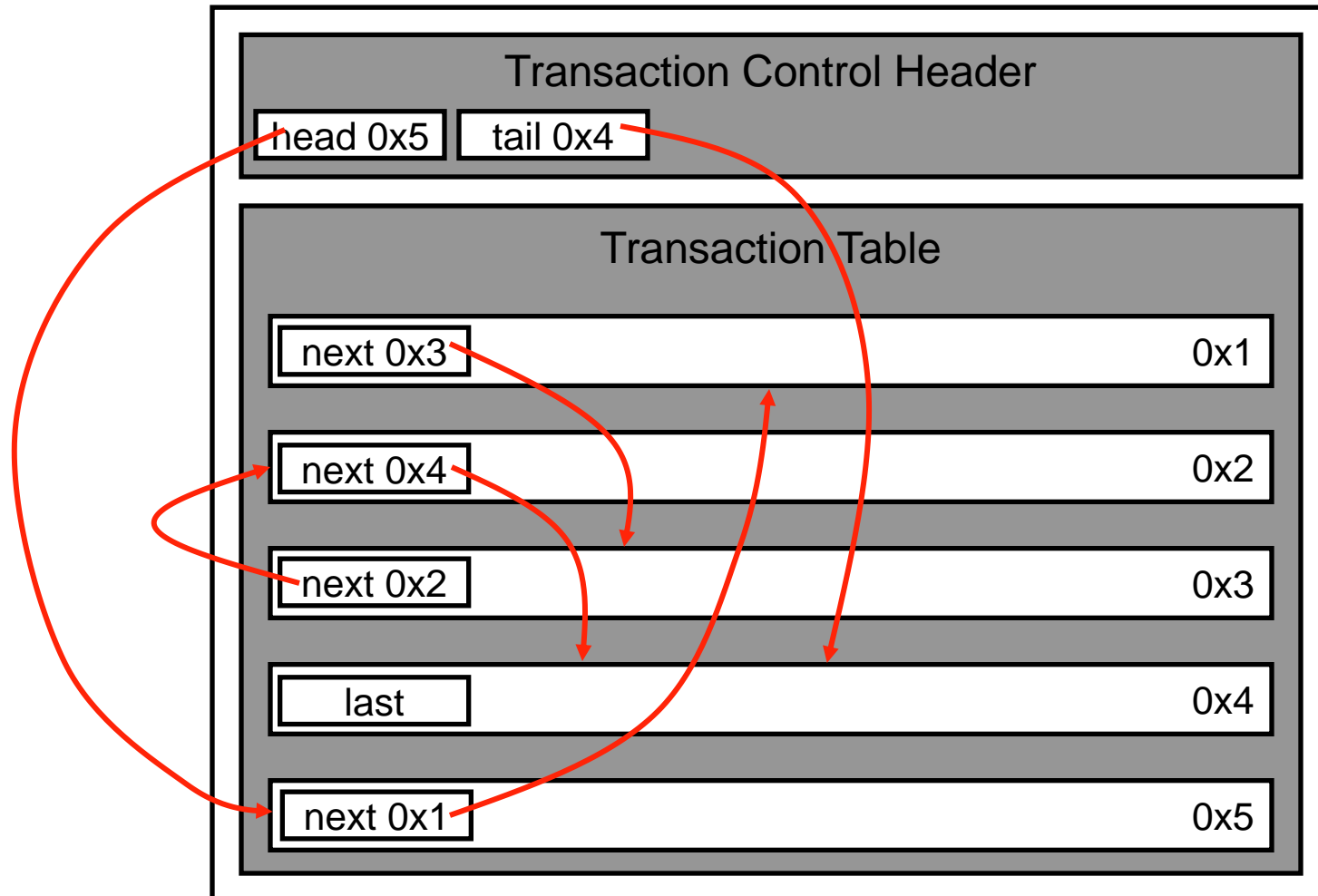
# Undo Chain



# Transaction Table Chain




## Undo Header



# Agenda

- Concepts
- Transaction Layer Structures
- Undo Structures
- ITL Waits and Deadlocks

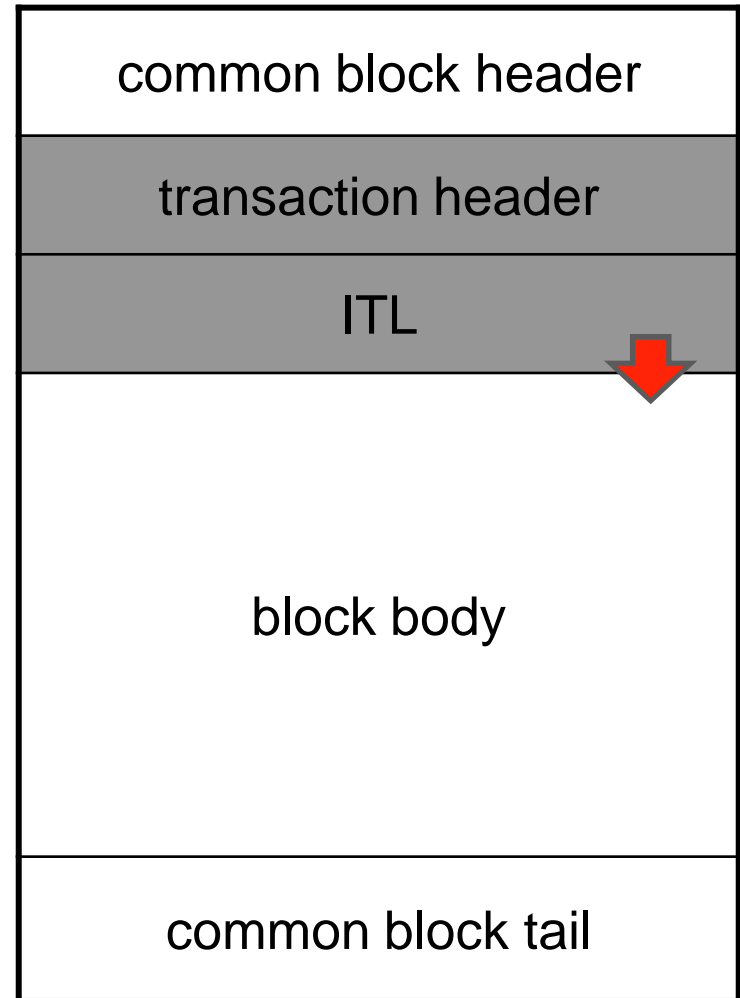


Data is always  
part of the game.

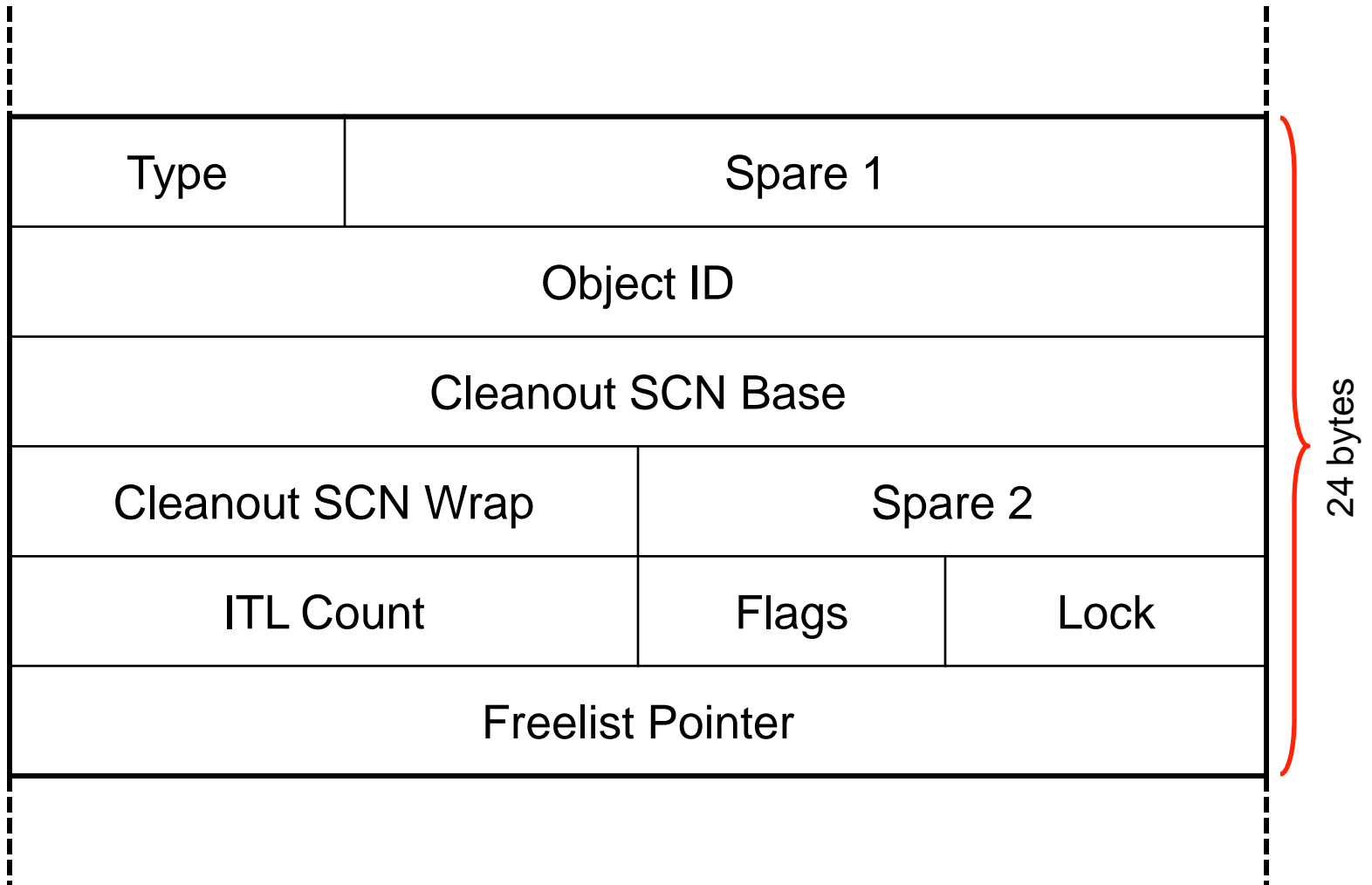
# Data Block



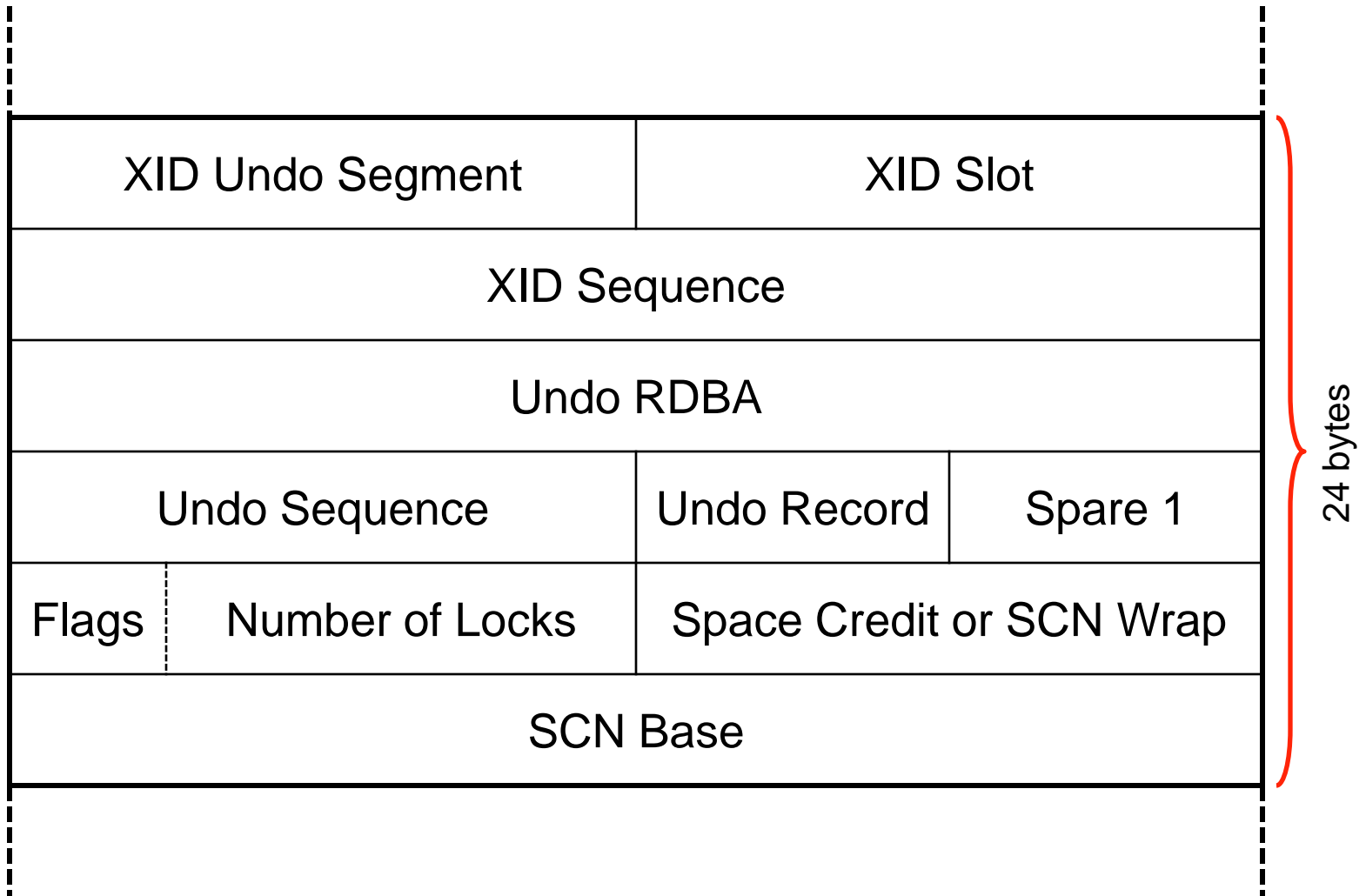
- Every data and index block below the high water mark, after the *common block header*, contains a *transaction header*.
- Every transaction header is followed by the *interested transaction list (ITL)*.
- Block type: 0x06



# Transaction Header




# Interested Transaction List Slot





# Agenda

- Concepts
- Transaction Layer Structures
- Undo Structures
- ITL Waits and Deadlocks

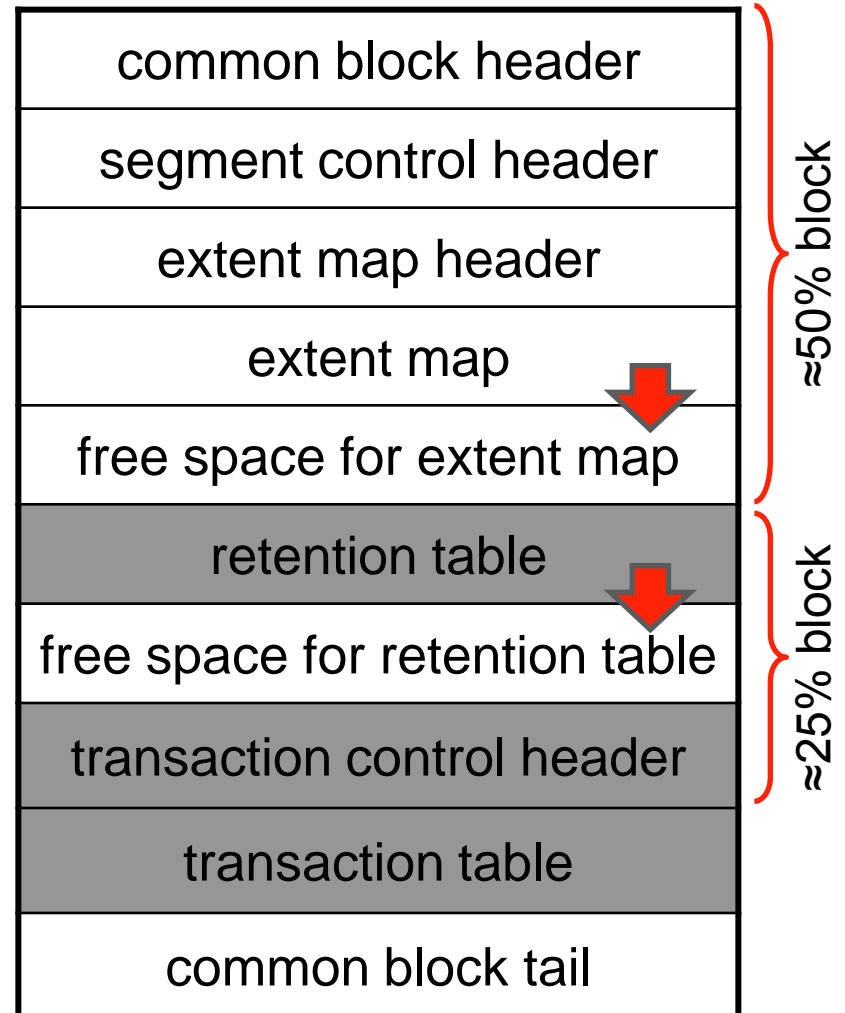


Data is always  
part of the game.

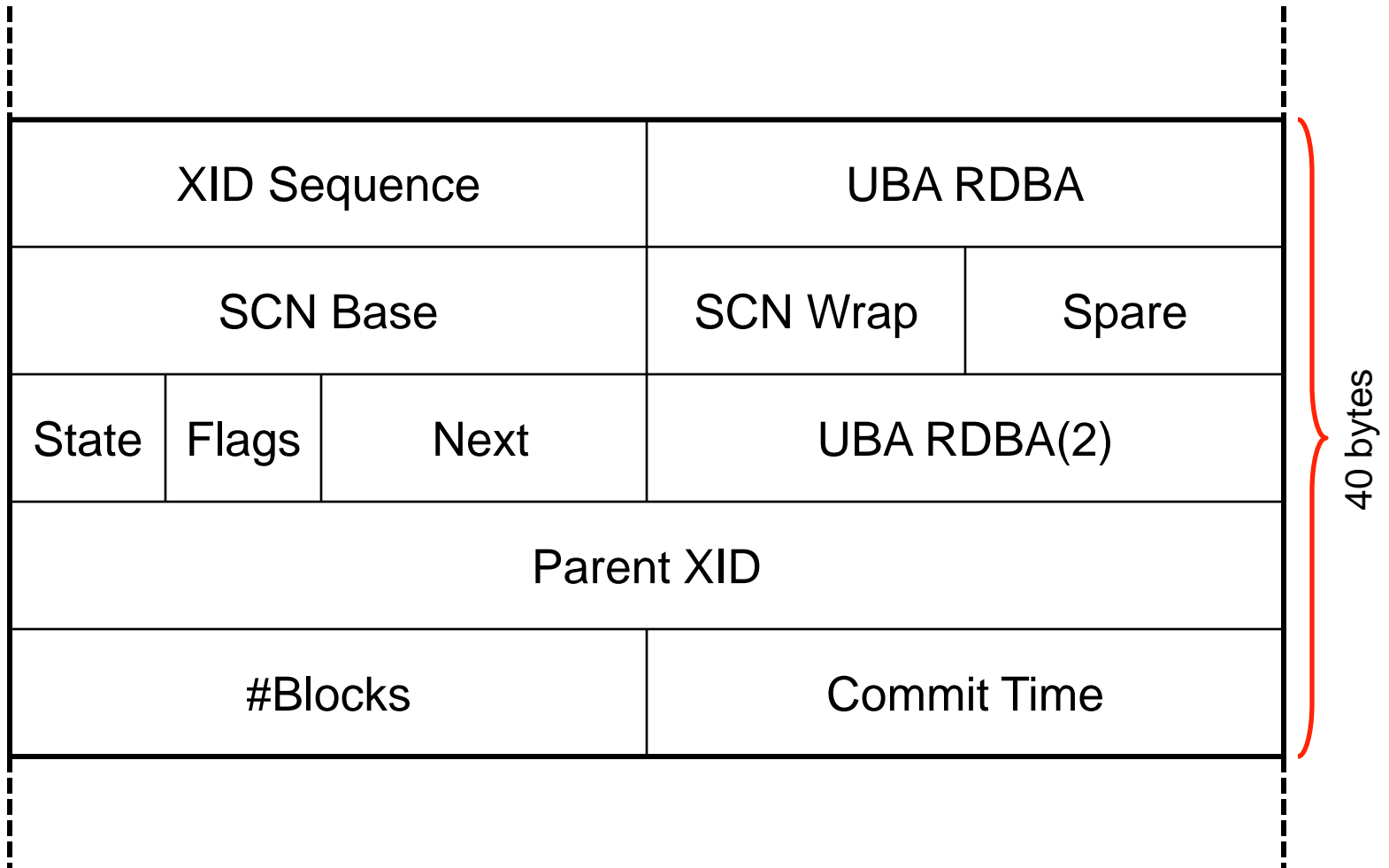
# System-Managed Undo Header



- The first half of the block is similar to the one of other segments.
- Only system-managed undo headers have a *retention table*.
- Block type: 0x26



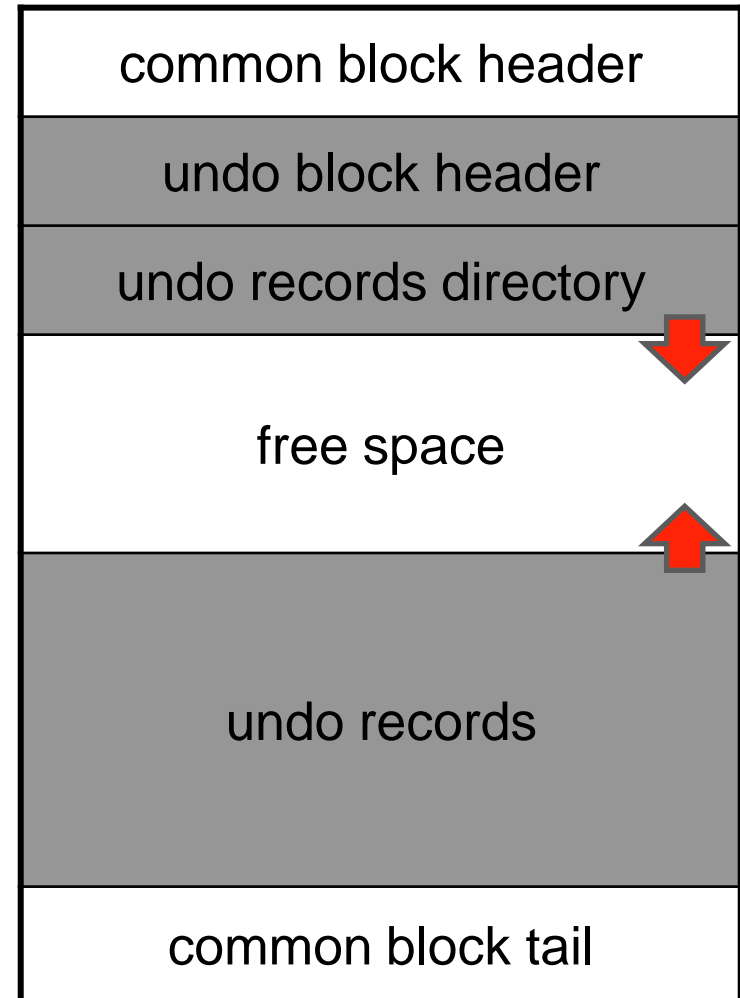
# Transaction Table Entry



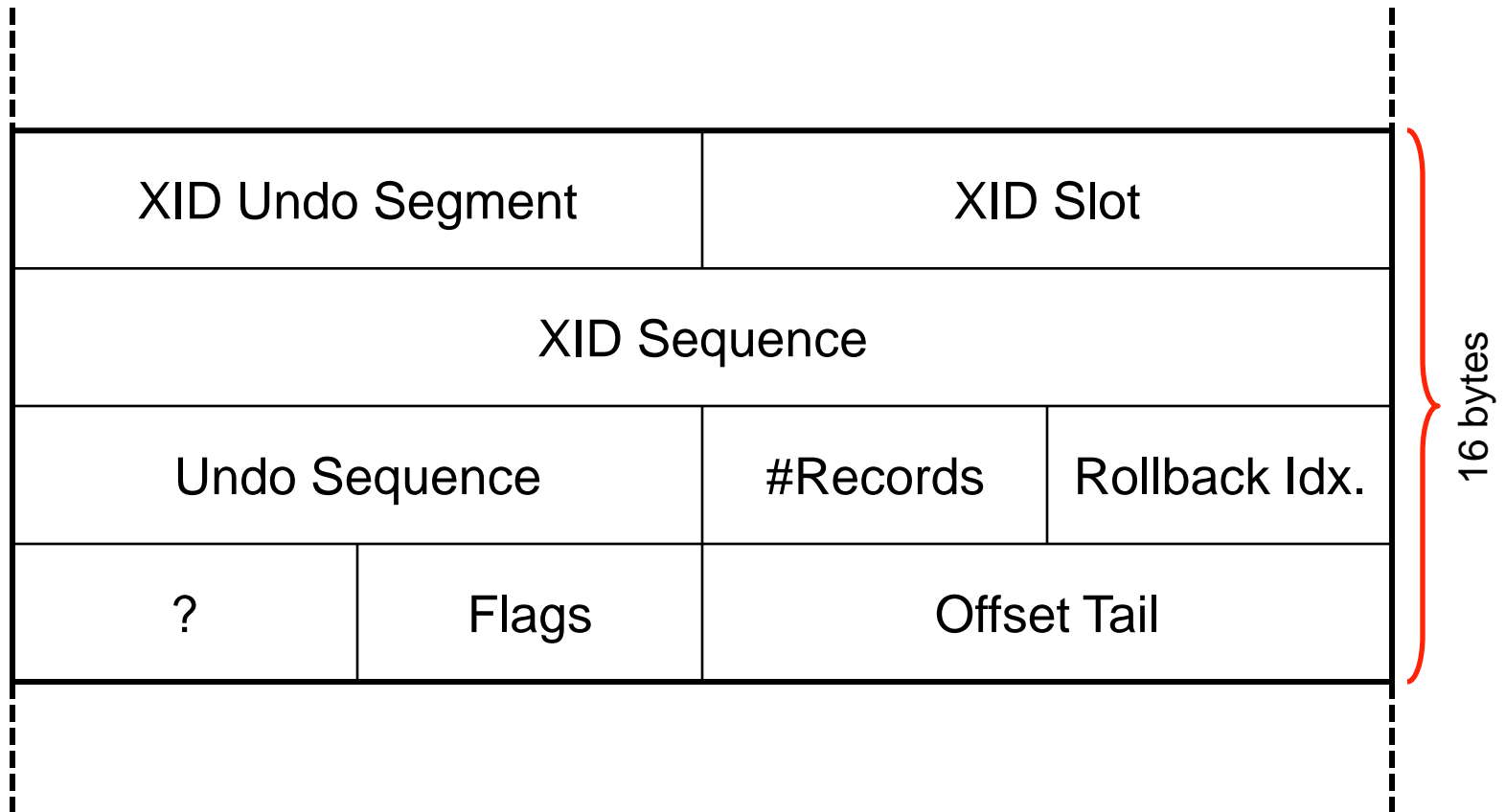
# Undo Block



- As for data blocks, there is a mapping between the logical position and the physical one.
- An undo block is used by a single active transaction concurrently.
- Block type: 0x02




# Undo Block Header



# Agenda

- Concepts
- Transaction Layer Structures
- Undo Structures
- ITL Waits and Deadlocks



Data is always  
part of the game.

# ITL Waits (1)



When a transaction is not able to get an ITL slot, it has to wait that one becomes free.

- This is called an *ITL wait*.
- The session waits on a specific ITL, not on the first being free!

Identification:

- “enq: TX - allocate ITL entry” event in V\$SESSION\_WAIT view
- Row in DBA\_WAITERS view
  - Lock type = Transaction (TX)
  - Mode held = Exclusive (X)
  - Mode requested = Share (S)
- “ITL waits” statistic in V\$SEGMENT\_STATISTICS view

# ITL Waits (2)



To avoid ITL waits there are three possibilities:

- Increase INITRANS
- Increase PCTFREE
- Decrease concurrency



# ITL Deadlocks



ITL waits can result in deadlocks.

The database engine raises an ORA-00060 to one of the waiting sessions.

Identification:

- Consult trace file generated for ORA-00060
  - Lock type = Transaction (TX)
  - Blocker mode = Exclusive (X)
  - Waiter mode = Share (S)
  - One or both sessions waits on “no row”

The only way to avoid ITL deadlocks is to avoid ITL waits.

# Core Messages



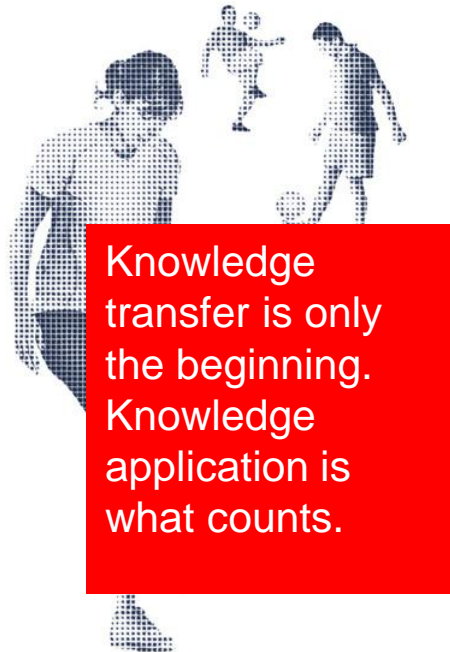
Every locked row points to an ITL slot.

An ITL slot references the undo data through an XID and an UBA.

At commit time a block cleanout should be performed.

Block cleanouts are delayed when more than few blocks are modified.

When not enough ITL slots are available, waits or even deadlocks are experienced.



Knowledge transfer is only the beginning. Knowledge application is what counts.

# Bibliography



Adams, Steve, “Oracle Internals and Advanced Performance Tuning.” Miracle Master Class, 2003.

Antognini, Christian, “Oracle Data Storage Internals.” Trivadis Training, 2005-2010.

# ■ ■ ■ Trivadis @ DOAG: 3<sup>th</sup> Floor



?

[www.trivadis.com](http://www.trivadis.com)

**trivadis**  
makes IT easier. ■ ■ ■



Basel

Bern

Lausanne

Zurich

Düsseldorf

Frankfurt/M.

Freiburg i. Br.

Hamburg

Munich

Stuttgart

Vienna