An Analysis of Ext4 for Digital Forensics

By

Kevin Fairbanks

Presented At

The Digital Forensic Research Conference

DFRWS 2012 USA Washington, DC (Aug 6th - 8th)

An Analysis of Ext4 for Digital Forensics

Kevin D. Fairbanks, PhD

DFRWS 2012

August 8th, 2012

APL
The Johns Hopkins University
APPLIED PHYSICS LABORATORY

Motivation and Objectives

Why is Ext4 important?

- **Motivation**
  - Default file system for newer Linux Installations
  - Android moving from YAFFS2 to Ext4
  - BtrFS almost ready

- **Objectives**
  - Comprehensive low-level study
    - Data persistence
    - New on-disk structures
    - Audience: Forensic Tool Makers and Analyst
  - Sleuthkit Extension

Related Work

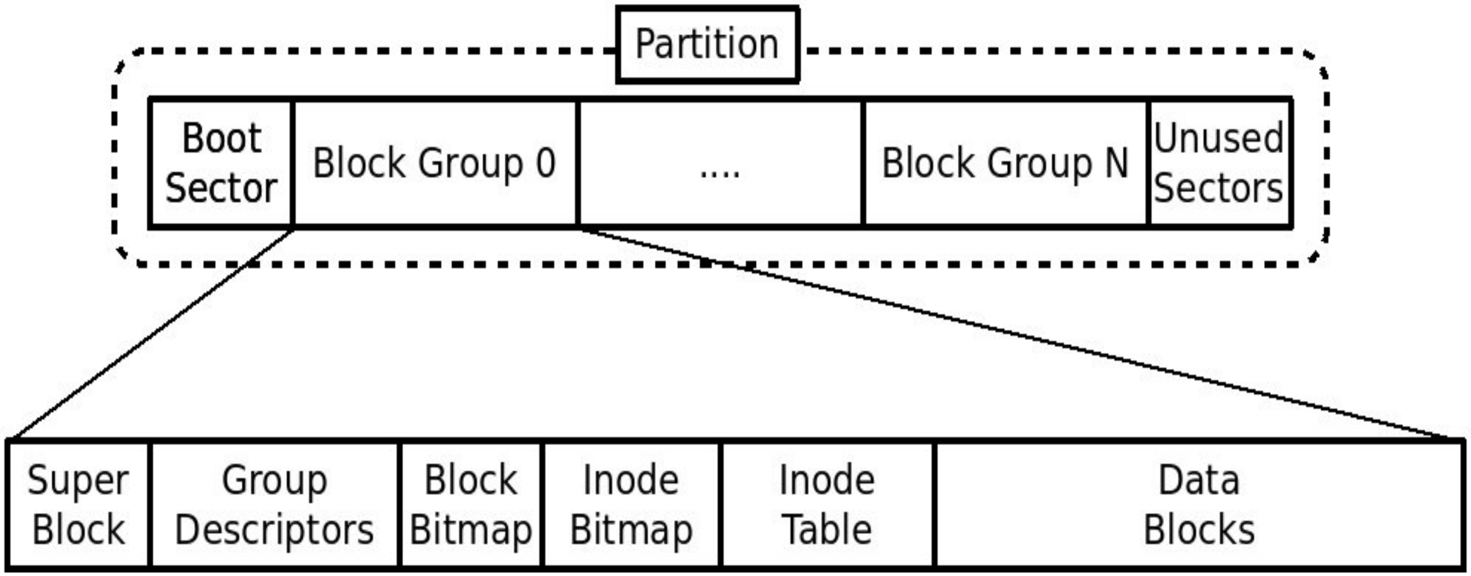
- **Ext4: bit by bit**
  - Hal Pomeranz
- **TSK Patches**
  - Willi Ballenthin
- **Forensic Implications of Ext4**
  - Kevin Fairbanks

Overview

- **Ext2/3 Primer**
- **Ext4**
  - Features
  - Scaling
  - Topology
  - Reliability
- **Forensic Implications**
- **TSK Ext4 Screenshots**

Ext2/3 Primer

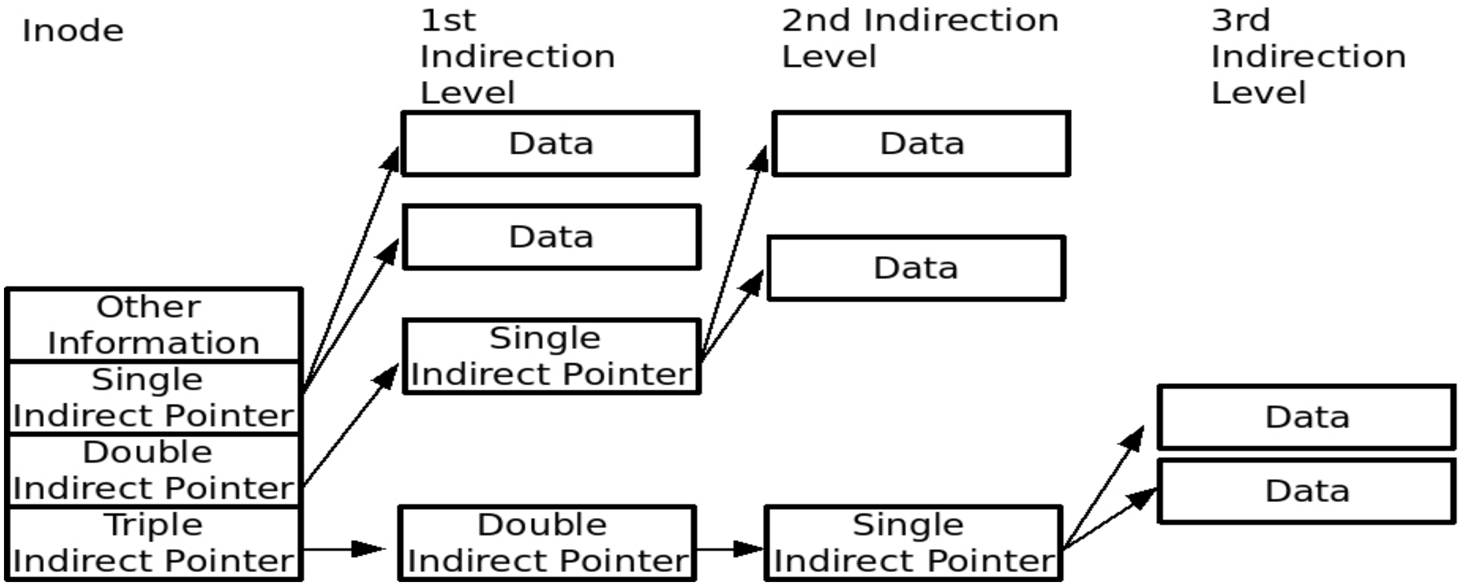
File System Layout



- FS divided into Block Groups
- Each Block Group contains FS meta-data
- Super Block and Group Descriptors may not be in every Block Group

Ext2/3 Primer

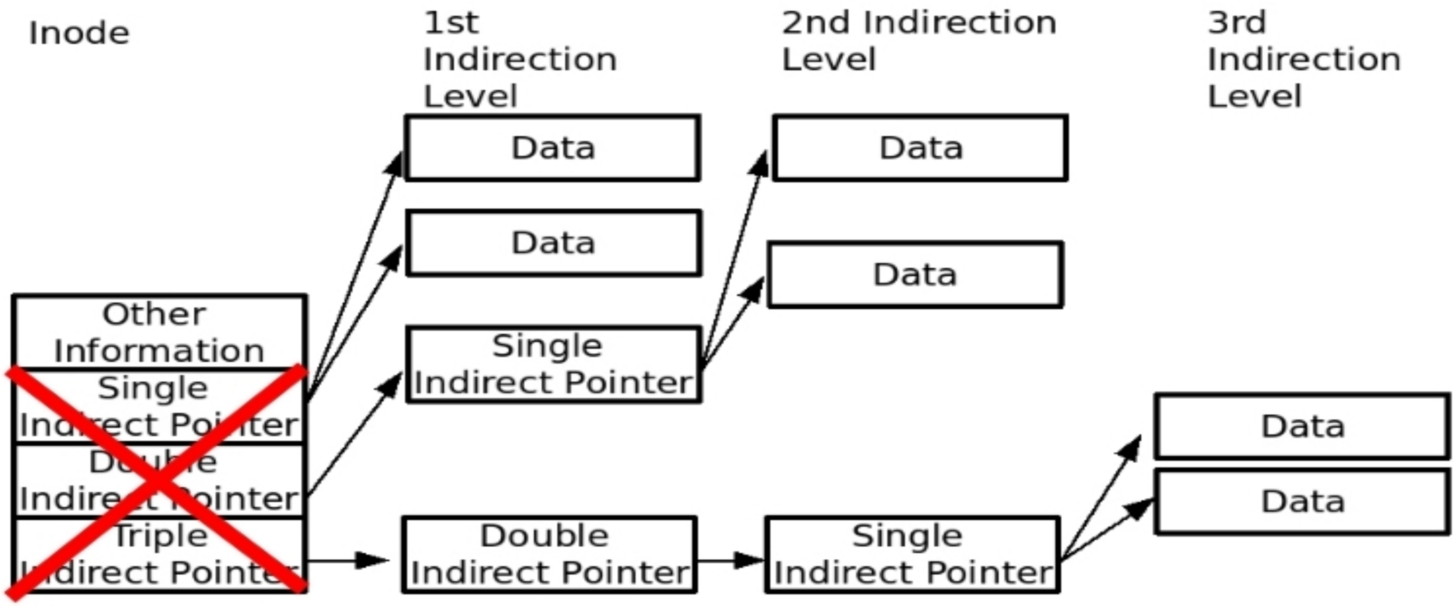
Data Mapping



- **File data mapped to inode using well known pointer system**

Ext2/3 Primer

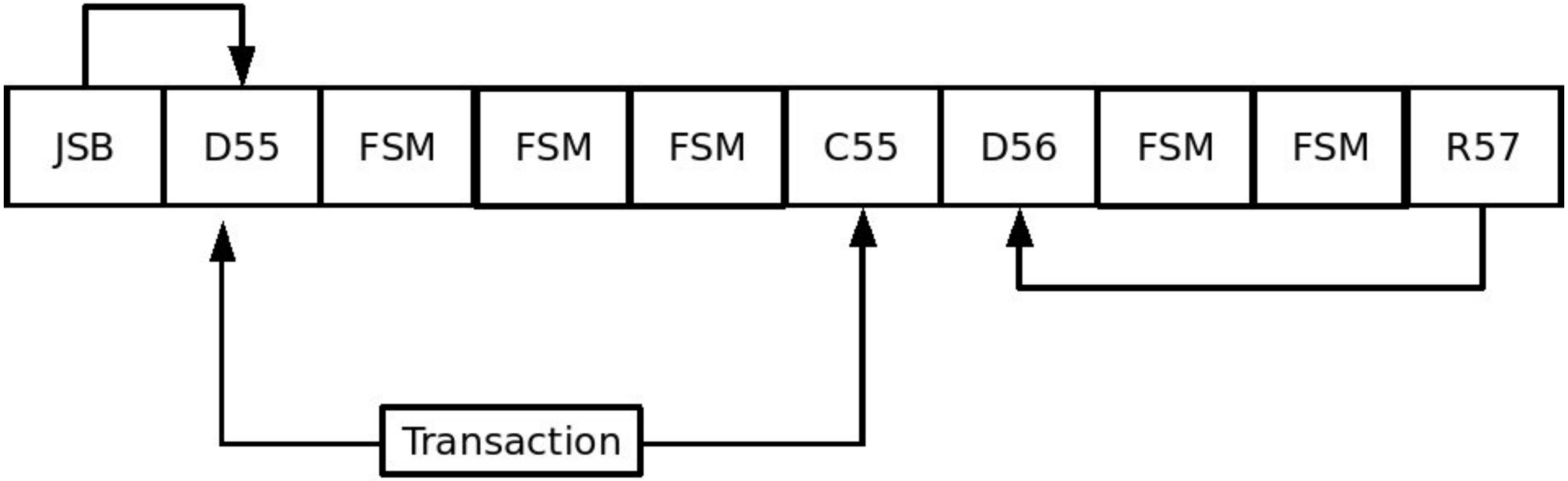
Data Mapping



- **File data mapped to inode using well known pointer system**
- **Deletion in Ext3 zeros inode resident pointers**

Ext2/3 Primer

Ext3 Journal



- **Transaction based cyclic log**
- **By default, metadata written to journal**
- **Uncommitted transactions revoked**
- **Two stage process**

Ext2/3 Primer

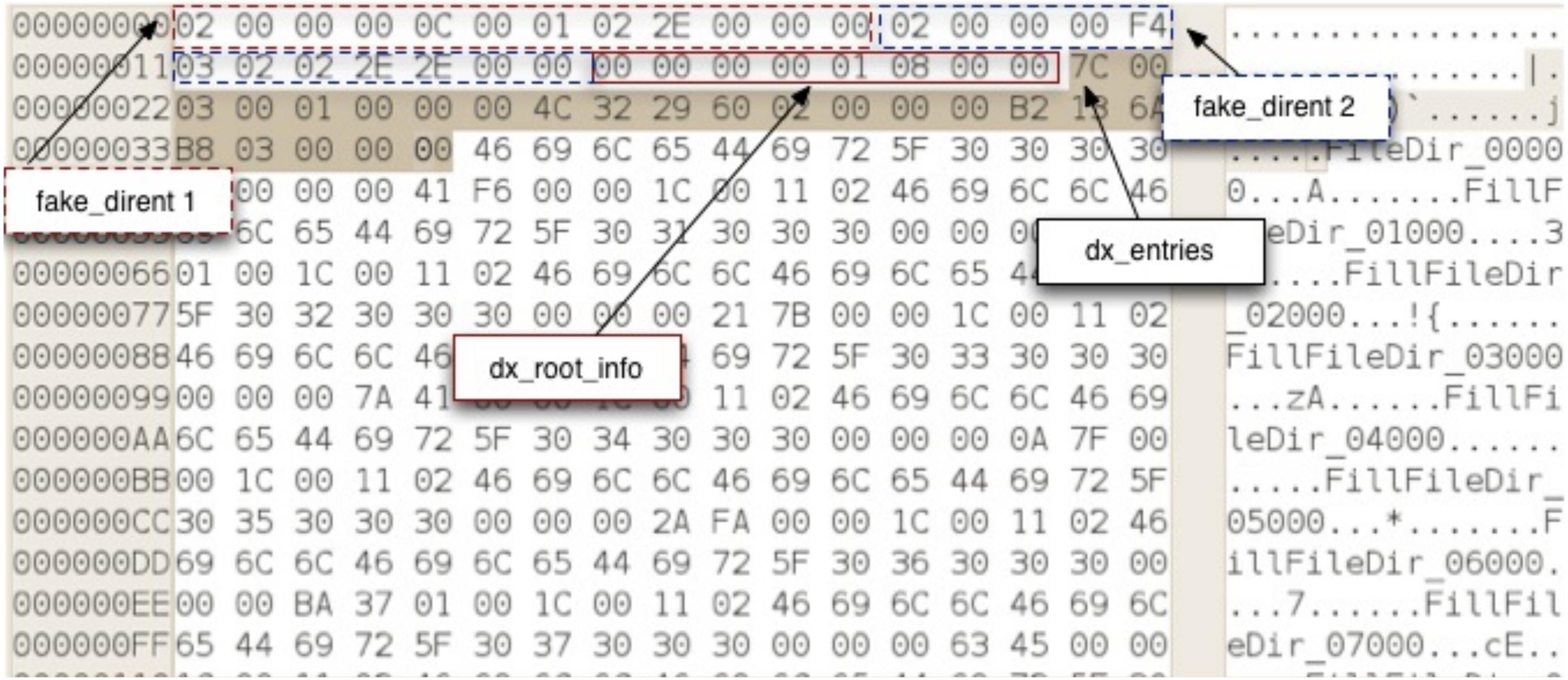
Directory Indexing

- **Optional in Ext2/3, but default in Ext4**
- **Constant depth H-tree used vs linked list**
  - Hash of filename and seed located in super block
  - Maximum depth – 2 levels
  - Leaf blocks are linked lists of directory entries
  - CRC32 checksum at end of each block

Ext2/3 Primer

Directory Indexing

- **Root Node Example**

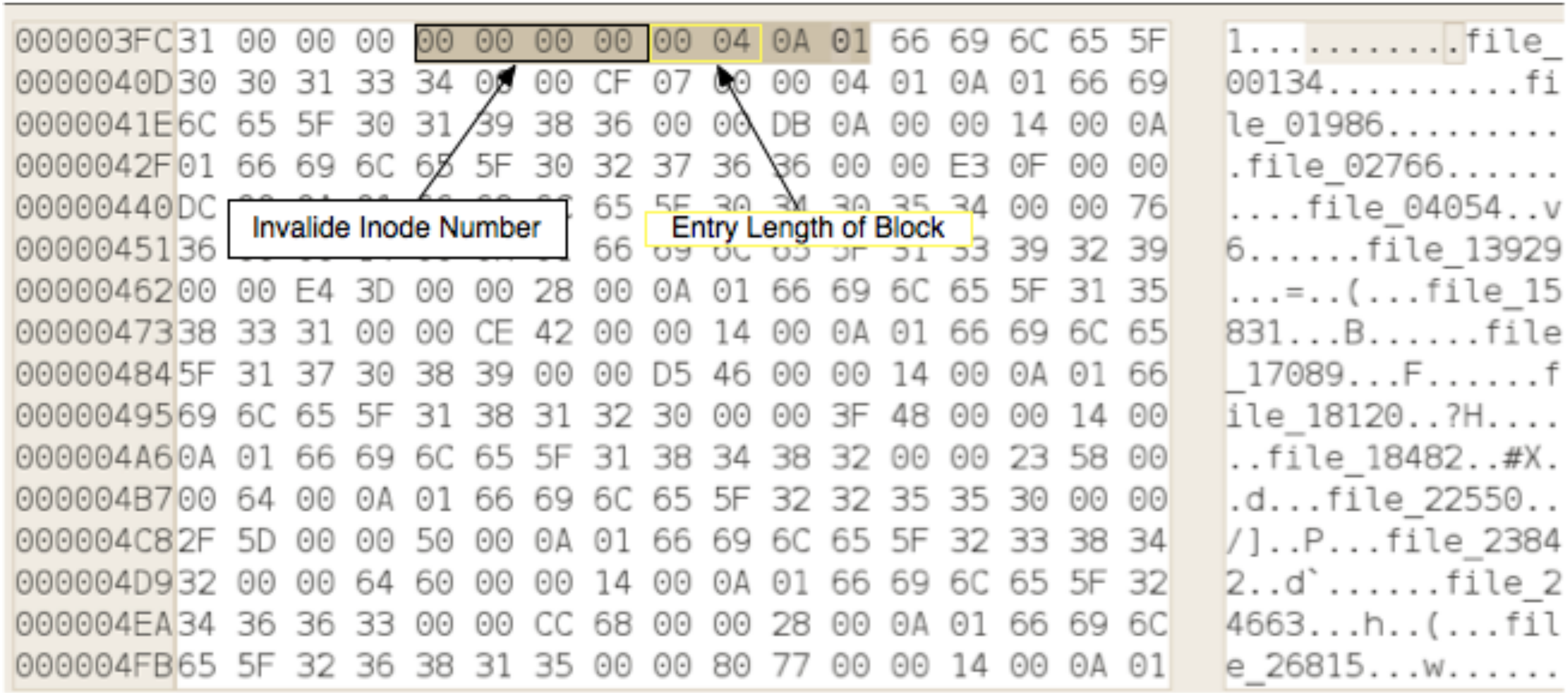


```
00000000 02 00 00 00 0C 00 01 02 2E 00 00 00 02 00 00 00 F4  .................
00000011 03 02 02 2E 2E 00 00 00 00 00 00 01 08 00 00 7C 00  ..........|.
00000022 03 00 01 00 00 00 4C 32 29 60 02 00 00 00 B2 1B 6A        )`......j
00000033 B8 03 00 00 00 46 69 6C 65 44 69 72 5F 30 30 30 30  .....FileDir_0000
            00 00 00 41 F6 00 00 1C 00 11 02 46 69 6C 6C 46  0...A........FillF
               6C 65 44 69 72 5F 30 31 30 30 30 00 00 00       eDir_01000....3
00000066 01 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44            ....FillFileDir
00000077 5F 30 32 30 30 30 00 00 00 21 7B 00 00 1C 00 11 02  _02000...!{......
00000088 46 69 6C 6C 46             69 72 5F 30 33 30 30 30  FillFileDir_03000
00000099 00 00 00 7A 41             11 02 46 69 6C 6C 46 69  ...zA......FillFi
000000AA 6C 65 44 69 72 5F 30 34 30 30 30 00 00 00 0A 7F 00  leDir_04000......
000000BB 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F  .....FillFileDir_
000000CC 30 35 30 30 30 00 00 00 2A FA 00 00 1C 00 11 02 46  05000...*.......F
000000DD 69 6C 6C 46 69 6C 65 44 69 72 5F 30 36 30 30 30 00  illFileDir_06000.
000000EE 00 00 BA 37 01 00 1C 00 11 02 46 69 6C 6C 46 69 6C  ...7......FillFil
000000FF 65 44 69 72 5F 30 37 30 30 30 00 00 00 63 45 00 00  eDir_07000...cE..
```

Ext2/3 Primer

Directory Indexing

- **Index Node Example**



```
000003FC31 00 00 00 00 00 00 00 00 04 0A 01 66 69 6C 65 5F   1...........file_
0000040D30 30 31 33 34 00 00 CF 07 00 00 04 01 0A 01 66 69   00134.........fi
0000041E6C 65 5F 30 31 39 38 36 00 00 DB 0A 00 00 14 00 0A   le_01986.........
0000042F01 66 69 6C 65 5F 30 32 37 36 36 00 00 E3 0F 00 00   .file_02766......
00000440DC 00 0A 01 66 69 6C 65 5F 30 34 30 35 34 00 00 76   ....file_04054..v
0000045136 00 00 [illegible] 00 0A 01 66 69 6C 65 5F 31 33 39 32 39   6......file_13929
0000046200 00 E4 3D 00 00 28 00 0A 01 66 69 6C 65 5F 31 35   ...=..(...file_15
0000047338 33 31 00 00 CE 42 00 00 14 00 0A 01 66 69 6C 65   831...B......file
000004845F 31 37 30 38 39 00 00 D5 46 00 00 14 00 0A 01 66   _17089...F......f
0000049569 6C 65 5F 31 38 31 32 30 00 00 3F 48 00 00 14 00   ile_18120..?H....
000004A60A 01 66 69 6C 65 5F 31 38 34 38 32 00 00 23 58 00   ..file_18482..#X.
000004B700 64 00 0A 01 66 69 6C 65 5F 32 32 35 35 30 00 00   .d...file_22550..
000004C82F 5D 00 00 50 00 0A 01 66 69 6C 65 5F 32 33 38 34   /]..P...file_2384
000004D932 00 00 64 60 00 00 14 00 0A 01 66 69 6C 65 5F 32   2..d`.....file_2
000004EA34 36 36 33 00 00 CC 68 00 00 28 00 0A 01 66 69 6C   4663...h..(...fil
000004FB65 5F 32 36 38 31 35 00 00 80 77 00 00 14 00 0A 01   e_26815...w......
```

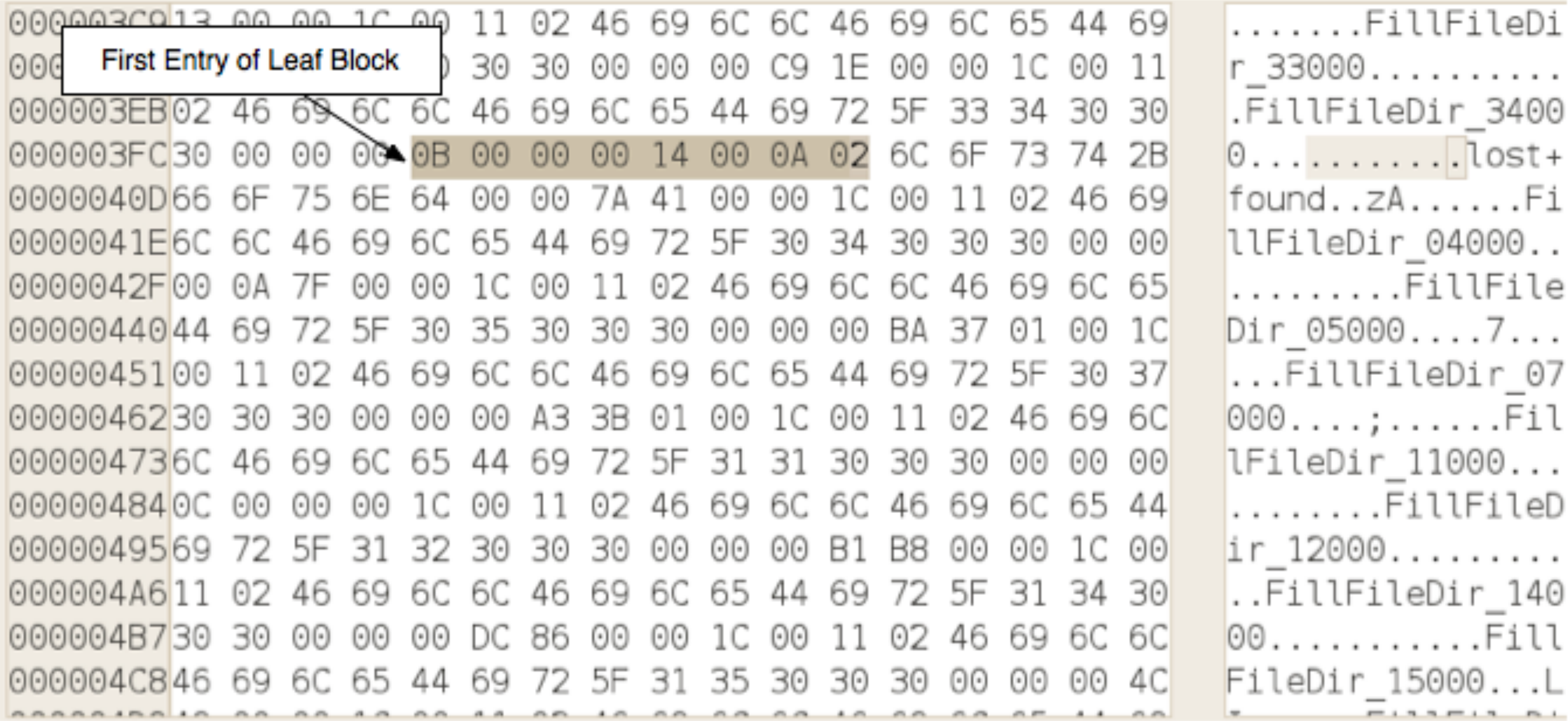
Invalide Inode Number

Entry Length of Block

Ext2/3 Primer

Directory Indexing

- **Leaf Node Example**



First Entry of Leaf Block

```
000003C9 13 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44 69   ........FillFileDi
000                         30 30 00 00 00 C9 1E 00 00 1C 00 11   r_33000...........
000003EB 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 33 34 30 30   .FillFileDir_3400
000003FC 30 00 00 00 0B 00 00 00 14 00 0A 02 6C 6F 73 74 2B   0............lost+
0000040D 66 6F 75 6E 64 00 00 7A 41 00 00 1C 00 11 02 46 69   found..zA......Fi
0000041E 6C 6C 46 69 6C 65 44 69 72 5F 30 34 30 30 30 00 00   llFileDir_04000..
0000042F 00 0A 7F 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65   .........FillFile
00000440 44 69 72 5F 30 35 30 30 30 00 00 00 BA 37 01 00 1C   Dir_05000....7...
00000451 00 11 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 30 37   ...FillFileDir_07
00000462 30 30 30 00 00 00 A3 3B 01 00 1C 00 11 02 46 69 6C   000....;......Fil
00000473 6C 46 69 6C 65 44 69 72 5F 31 31 30 30 30 00 00 00   lFileDir_11000...
00000484 0C 00 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44   ........FillFileD
00000495 69 72 5F 31 32 30 30 30 00 00 00 B1 B8 00 00 1C 00   ir_12000.........
000004A6 11 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 31 34 30   ..FillFileDir_140
000004B7 30 30 00 00 00 DC 86 00 00 1C 00 11 02 46 69 6C 6C   00...........Fill
000004C8 46 69 6C 65 44 69 72 5F 31 35 30 30 30 00 00 00 4C   FileDir_15000...L
```

Ext4 Features

Just Ext3 with extents, right?

- Flexible Block Groups
- Directory Hashing (Default)
- Extents
- Huge Files
- Persistent Preallocation
- Nanosecond Timestamps
- Journal Block Device 2
- More to come?

Ext4

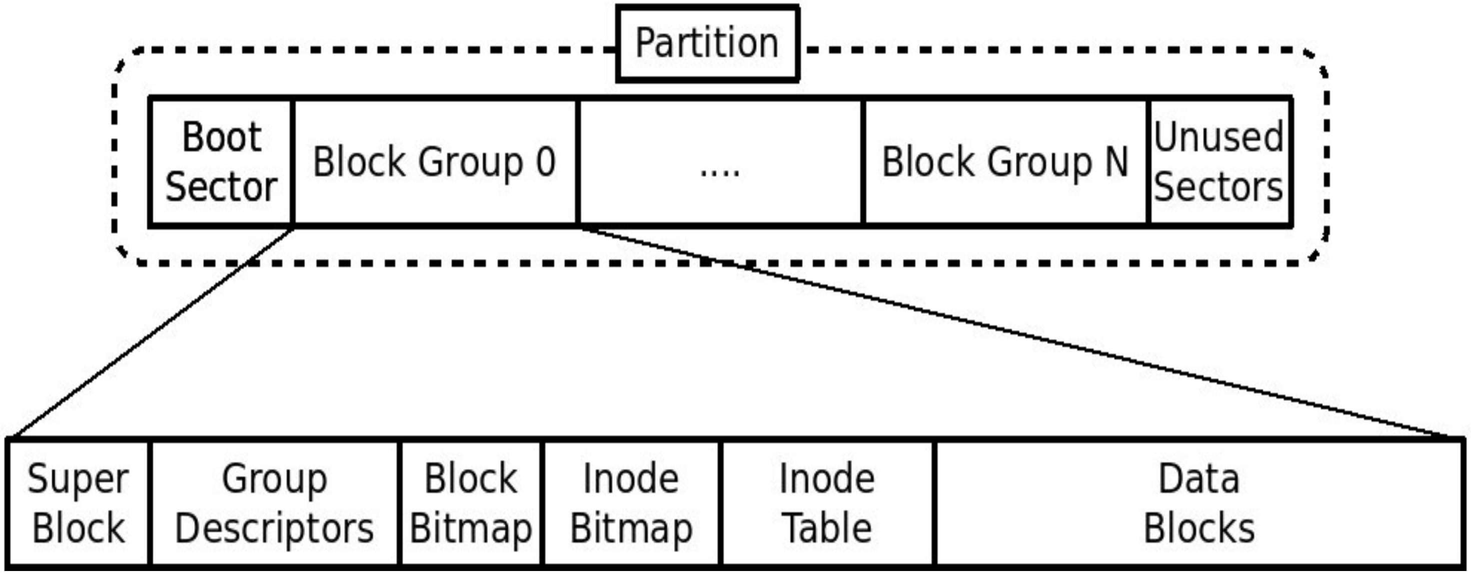
File System Scaling

- **Maximum file system size**
  - Ext3: 16 TB
    - 32 bit address space
  - Ext4: 1 EB = 10^3 PB = 10^6 TB*
    - 48 bit address space
- **Maximum file size**
  - Ext3: 2TB
    - 32 bit i_blocks field
  - Ext4: 16TB
    - HUGE_FILE flag means i_blocks is blocks not sectors
- **Max Files Per Directory**
  - Ext3: 32K
  - Ext4: Unlimited
    - Link Counter set to 1
    - Directory Indexing Used

*Despite the footnote in the paper

Ext4 Topology

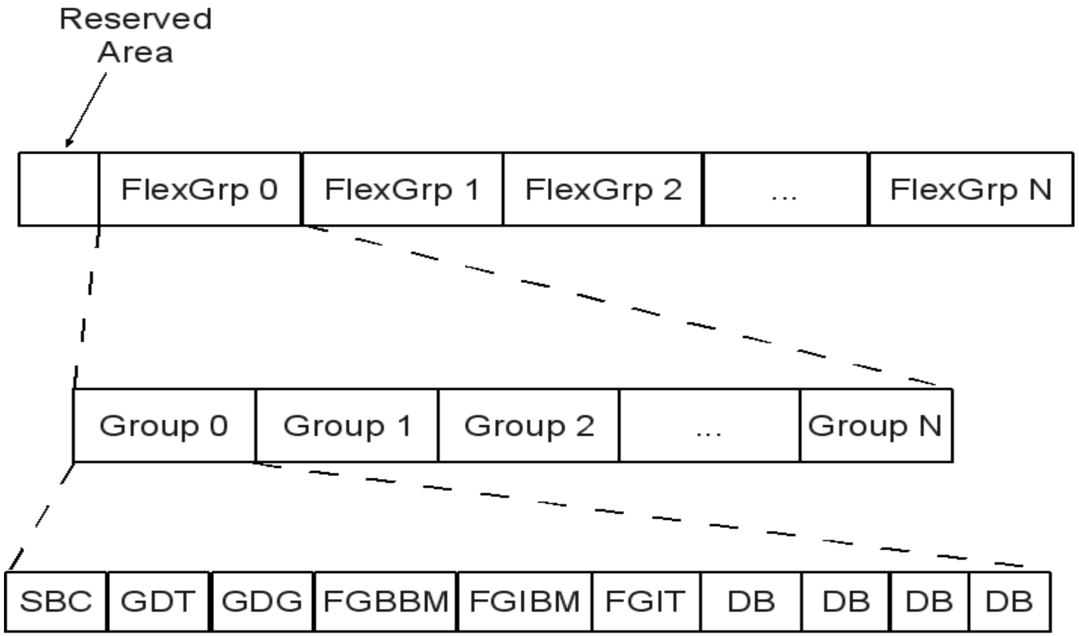
Ext2/3 File System Layout



- **FS divided into Block Groups**
- **Each Block Group contains FS meta-data**
- **Super Block and Group Descriptors may not be in every Block Group**

Ext4 Topology

Ext4 File System Layout



Reserved Area

FlexGrp 0 | FlexGrp 1 | FlexGrp 2 | ... | FlexGrp N

Group 0 | Group 1 | Group 2 | ... | Group N

SBC | GDT | GDG | FGBBM | FGIBM | FGIT | DB | DB | DB | DB

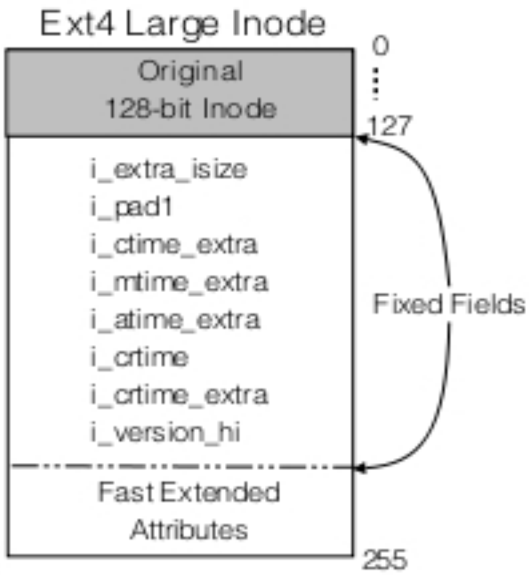
Key

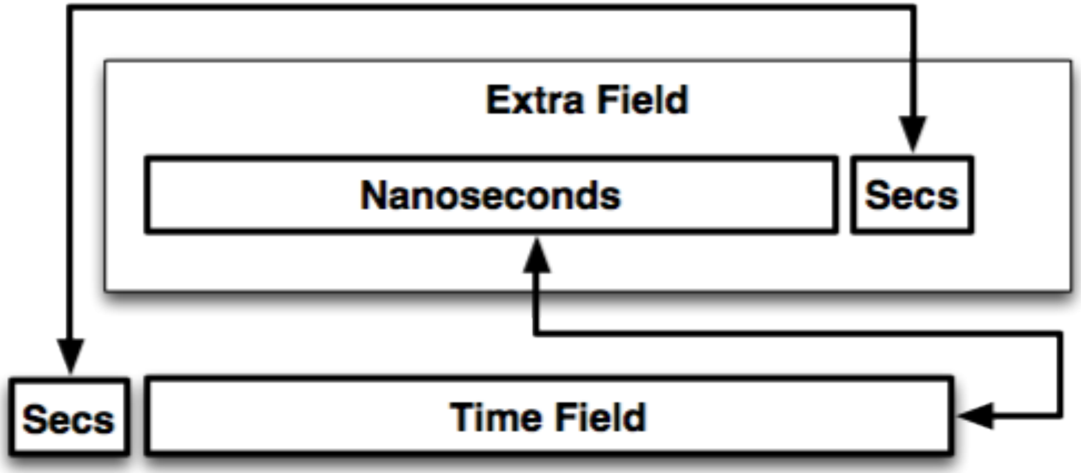
SBC	Super Block Copy
GDT	Grp Desc Table
GDG	Grp Desc Growth
FGBBM	FlxGrp Blk Bitmap
FGIBM	FlxGrp Inode Bitmap
FGIT	FlxGrp Inode Table
DB	Data Block

- **Block Groups combined into Flex Groups**
- **Metadata no longer resides inside a particular block group**
- **GDG blocks reserved for expansion**
- **Lazy initialization of bitmaps and inode tables (lazy_bg)**

Ext4 Topology

Nanosecond Timestamps

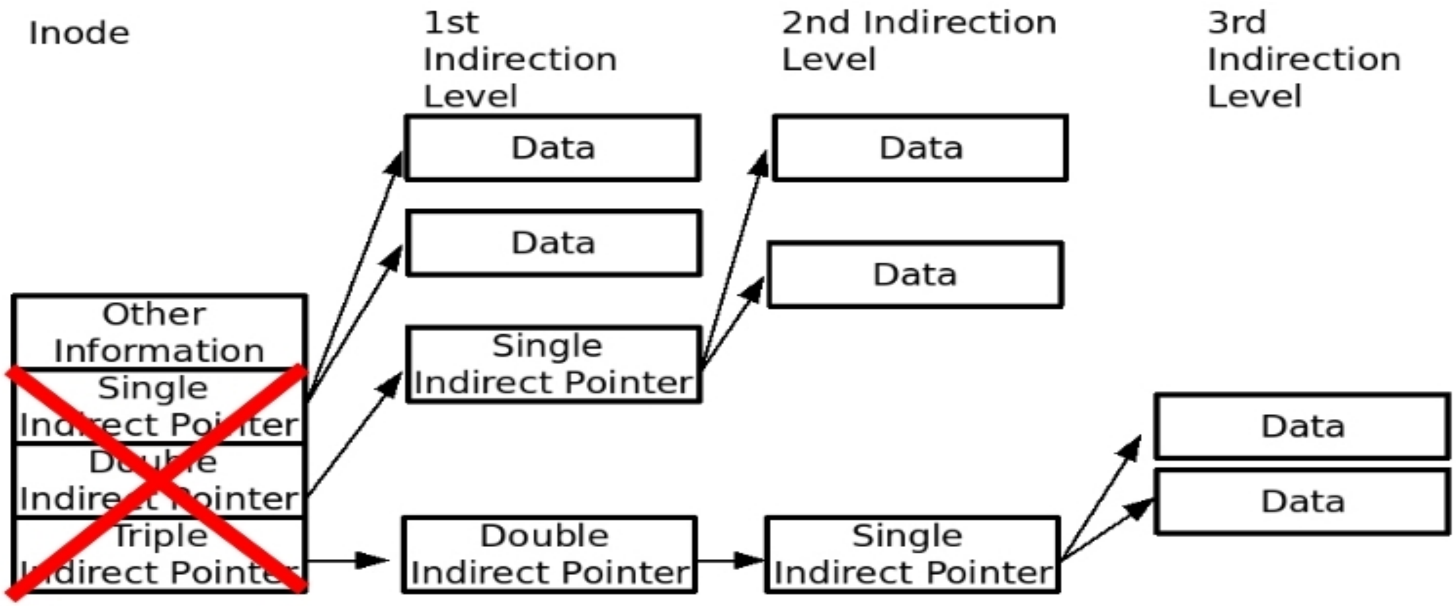




- MAC times get better resolution
- Creation timestamp introduced
- Deletion timestamp still has second resolution
- High 30 bits used for nanoseconds lower 2 bits extend timestamp

Ext4 Topology

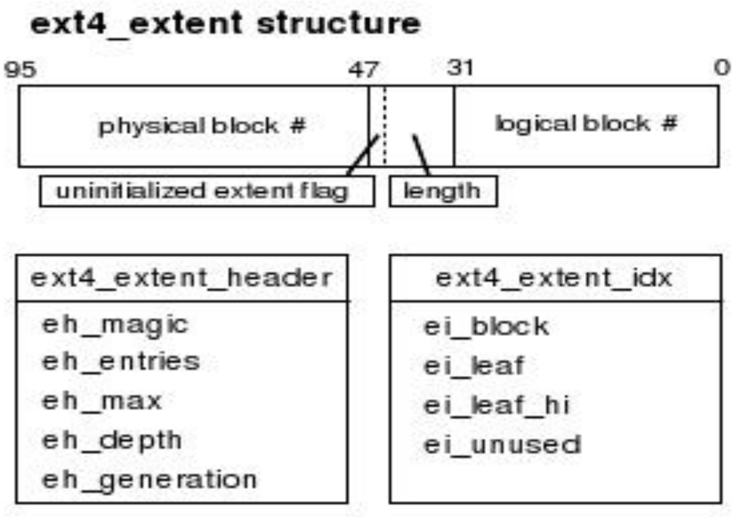
Ext2/3 Mapping

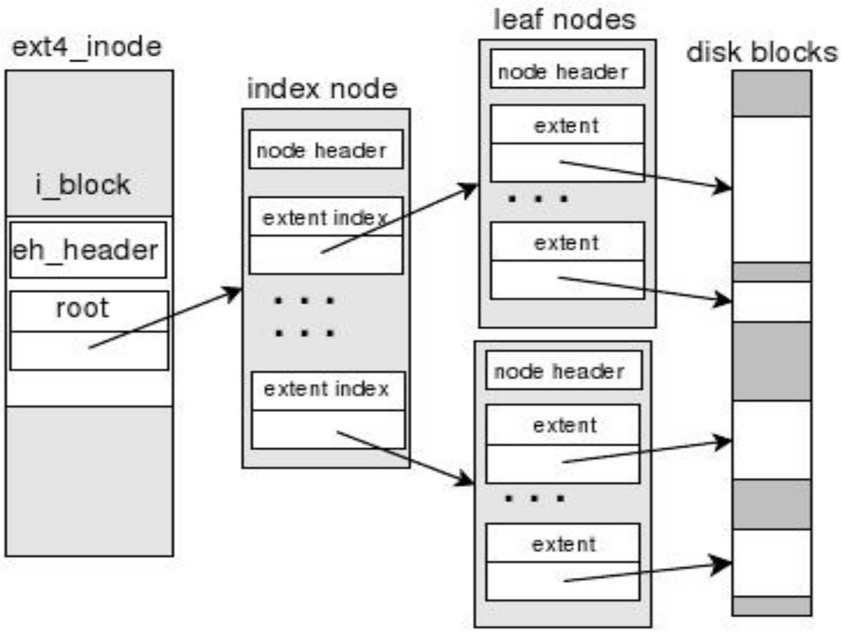


- **File data mapped to inode using well known pointer system**
- **Deletion in Ext3 zeros inode resident pointers**

Ext4 Topology

Ext4 Data Mapping

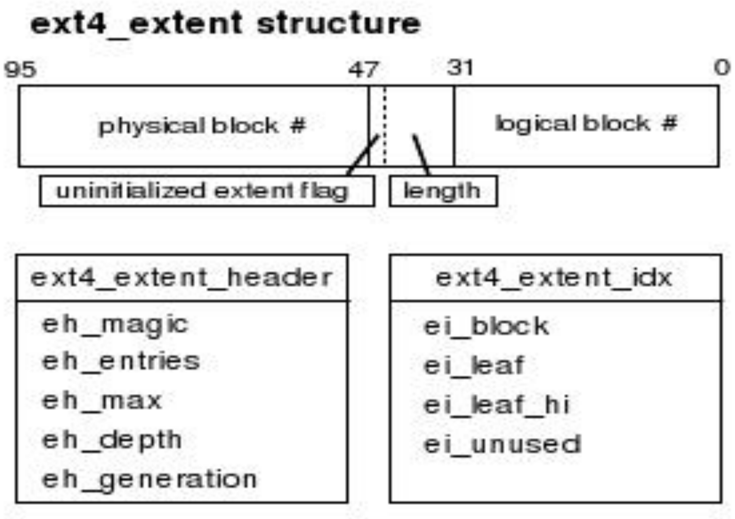




- Extents can reside in inode or form a tree
- Every level of tree has an extent header
- Persistent Preallocation

Ext4 Topology

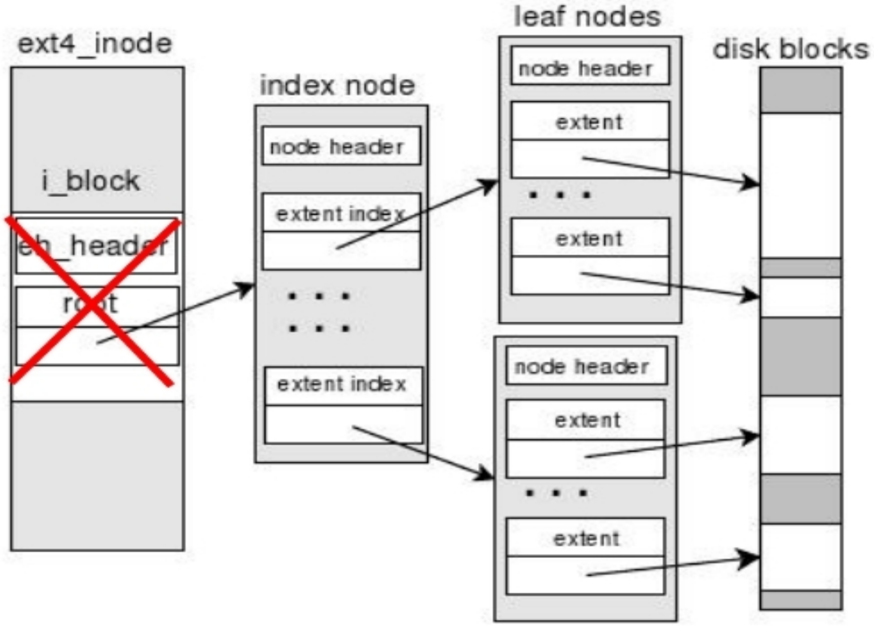
Ext4 Data Mapping



ext4_extent structure

ext4_extent_header
- eh_magic
- eh_entries
- eh_max
- eh_depth
- eh_generation

ext4_extent_idx
- ei_block
- ei_leaf
- ei_leaf_hi
- ei_unused

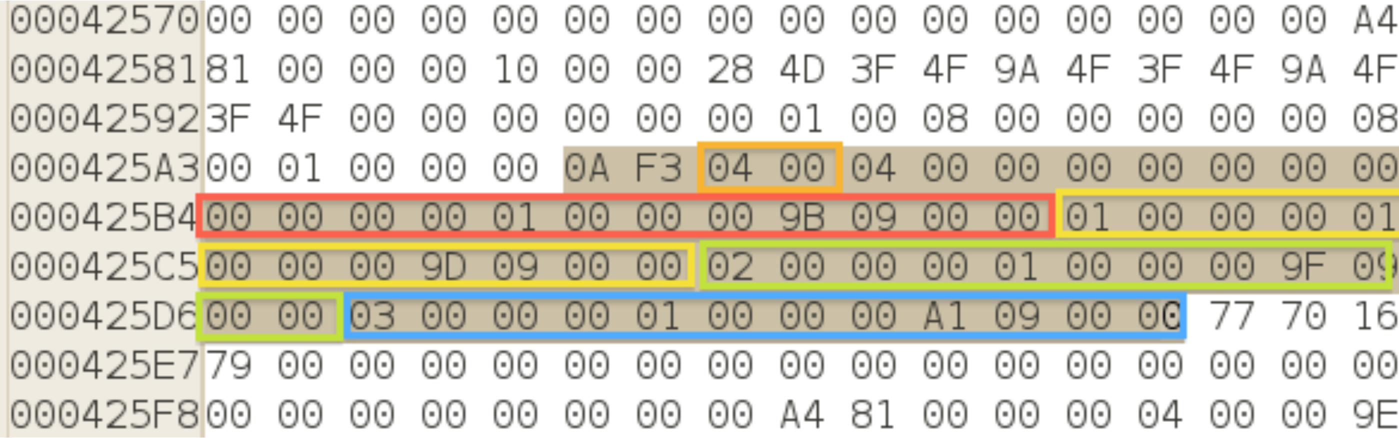


- **Zeroing of inode resident extents depends upon creation of extent tree**

Ext4 Topology

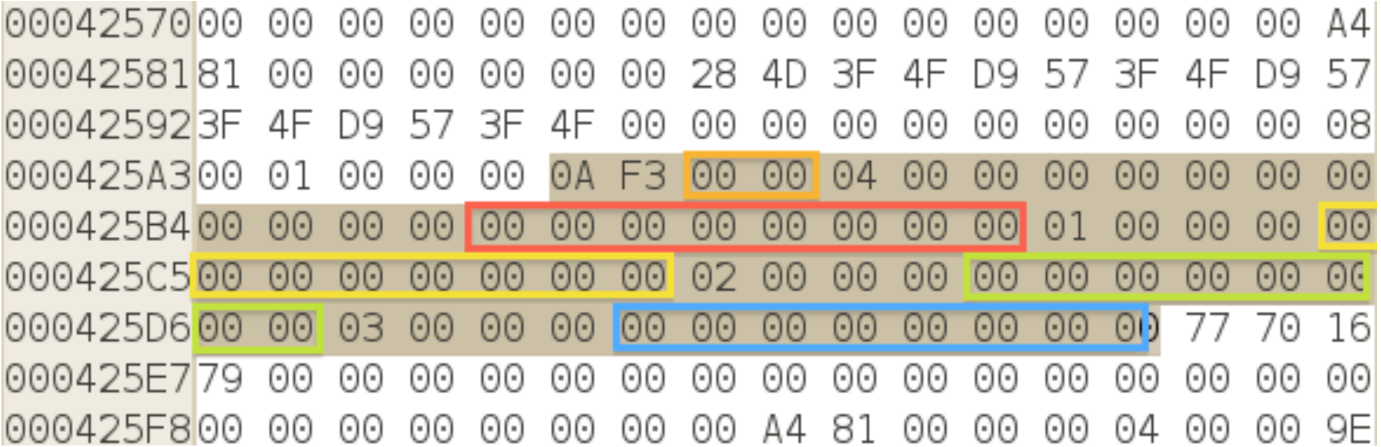
Extent Resident File Deletion

Before Deletion



```
00042570 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A4
00042581 81 00 00 00 10 00 00 28 4D 3F 4F 9A 4F 3F 4F 9A 4F
00042592 3F 4F 00 00 00 00 00 00 01 00 08 00 00 00 00 00 08
000425A3 00 01 00 00 00 0A F3 04 00 04 00 00 00 00 00 00 00
000425B4 00 00 00 00 01 00 00 00 9B 09 00 00 01 00 00 00 01
000425C5 00 00 00 9D 09 00 00 02 00 00 00 01 00 00 00 9F 09
000425D6 00 00 03 00 00 00 01 00 00 00 A1 09 00 00 77 70 16
000425E7 79 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000425F8 00 00 00 00 00 00 00 00 A4 81 00 00 00 04 00 00 9E
```

After Deletion

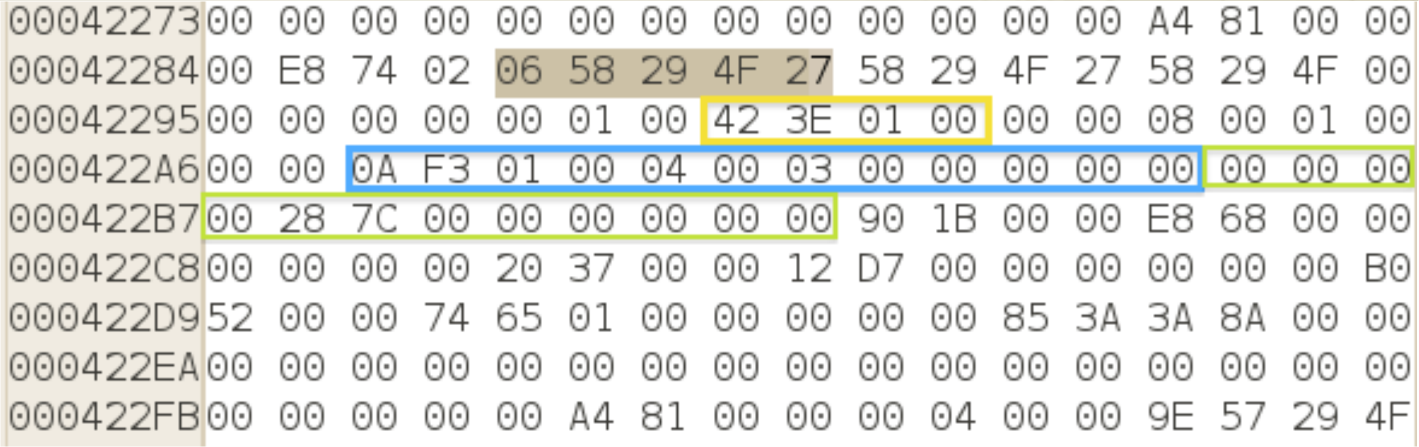


```
00042570 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 A4
00042581 81 00 00 00 00 00 00 28 4D 3F 4F D9 57 3F 4F D9 57
00042592 3F 4F D9 57 3F 4F 00 00 00 00 00 00 00 00 00 00 08
000425A3 00 01 00 00 00 0A F3 00 00 04 00 00 00 00 00 00 00
000425B4 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00
000425C5 00 00 00 00 00 00 00 02 00 00 00 00 00 00 00 00 00
000425D6 00 00 03 00 00 00 00 00 00 00 00 00 00 00 77 70 16
000425E7 79 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000425F8 00 00 00 00 00 00 00 00 A4 81 00 00 00 04 00 00 9E
```

Ext4 Topology

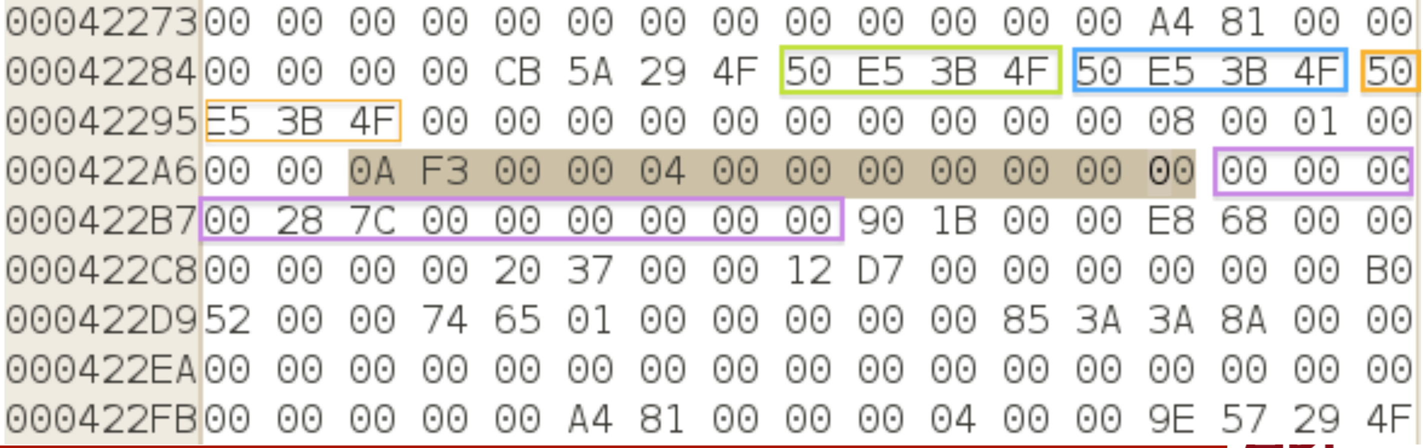
Extent Tree File Deletion

Before Deletion



```
00042273 00 00 00 00 00 00 00 00 00 00 00 00 00 A4 81 00 00
00042284 00 E8 74 02 06 58 29 4F 27 58 29 4F 27 58 29 4F 00
00042295 00 00 00 00 00 01 00 42 3E 01 00 00 00 08 00 01 00
000422A6 00 00 0A F3 01 00 04 00 03 00 00 00 00 00 00 00 00
000422B7 00 28 7C 00 00 00 00 00 00 90 1B 00 00 E8 68 00 00
000422C8 00 00 00 00 20 37 00 00 12 D7 00 00 00 00 00 00 B0
000422D9 52 00 00 74 65 01 00 00 00 00 00 85 3A 3A 8A 00 00
000422EA 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000422FB 00 00 00 00 00 A4 81 00 00 00 04 00 00 9E 57 29 4F
```

After Deletion



```
00042273 00 00 00 00 00 00 00 00 00 00 00 00 00 A4 81 00 00
00042284 00 00 00 00 CB 5A 29 4F 50 E5 3B 4F 50 E5 3B 4F 50
00042295 E5 3B 4F 00 00 00 00 00 00 00 00 00 00 08 00 01 00
000422A6 00 00 0A F3 00 00 04 00 00 00 00 00 00 00 00 00 00
000422B7 00 28 7C 00 00 00 00 00 00 90 1B 00 00 E8 68 00 00
000422C8 00 00 00 00 20 37 00 00 12 D7 00 00 00 00 00 00 B0
000422D9 52 00 00 74 65 01 00 00 00 00 00 85 3A 3A 8A 00 00
000422EA 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
000422FB 00 00 00 00 00 A4 81 00 00 00 04 00 00 9E 57 29 4F
```

Ext4 Reliability

- **Journal Block Device 2**
  - 64bit and 32bit systems
  - CRC32 checksum added to the commit block
  - Computed over all transaction blocks
  - Commit block written to journal in 1 step process

- **Group Descriptor Checksums**
  - CRC16
  - Verify inode count
  - Can skip over unused areas during e2fsck

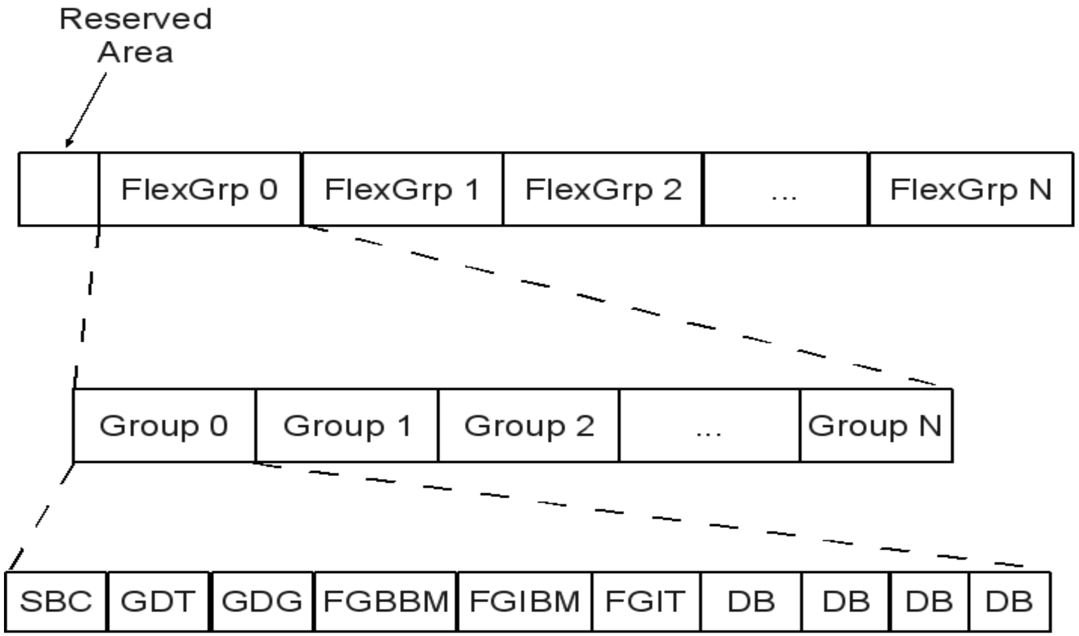
Forensic Implications I

- **Deleted Files**
  - Inode resident extent dependent upon tree creation
  - Extent index node not zeroed
  - Extent headers for file recovery

- **Metadata**
  - FS metadata statically located
  - File metadata mixed with normal blocks
  - IFF 256-byte Inode
    - Increased timestamp resolution
    - Creation Timestamp

Ext4 Topology

Ext4 File System Layout



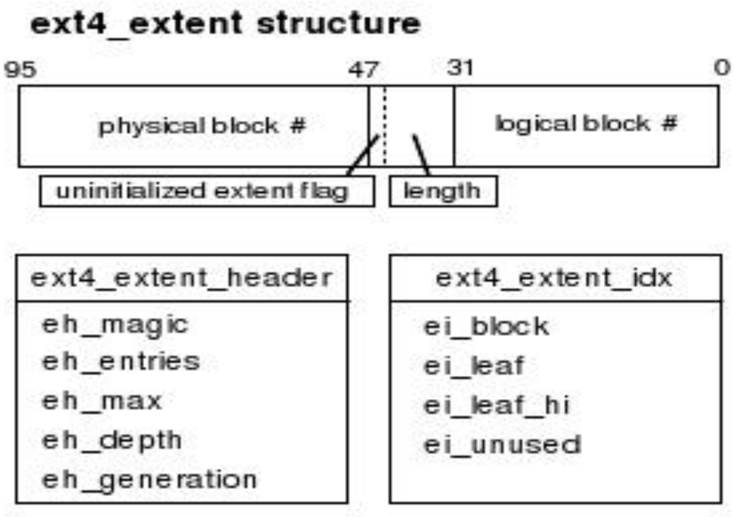
Key

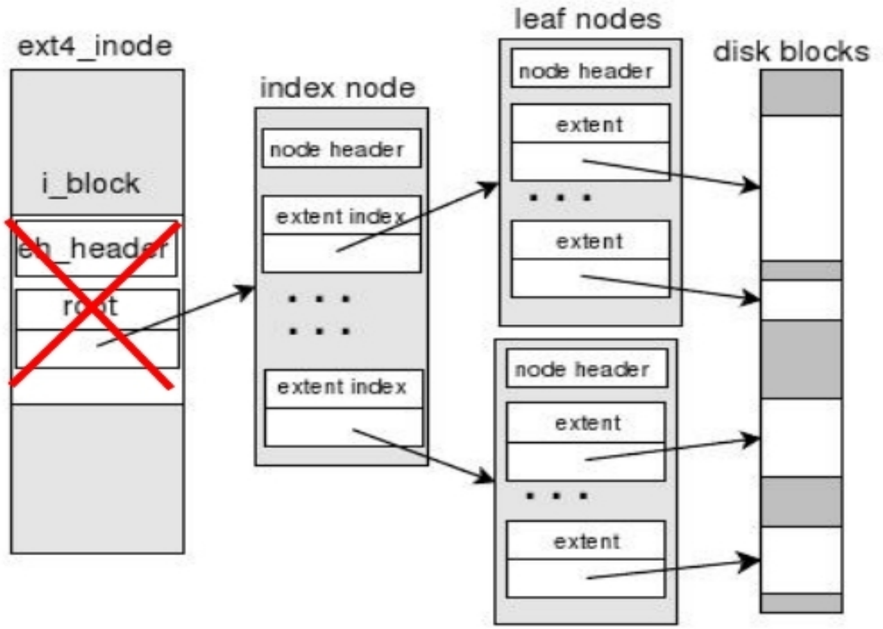
SBC	Super Block Copy
GDT	Grp Desc Table
GDG	Grp Desc Growth
FGBBM	FlxGrp Blk Bitmap
FGIBM	FlxGrp Inode Bitmap
FGIT	FlxGrp Inode Table
DB	Data Block

- **File system metadata structures do not move around**

Ext4 Topology

Ext4 Data Mapping





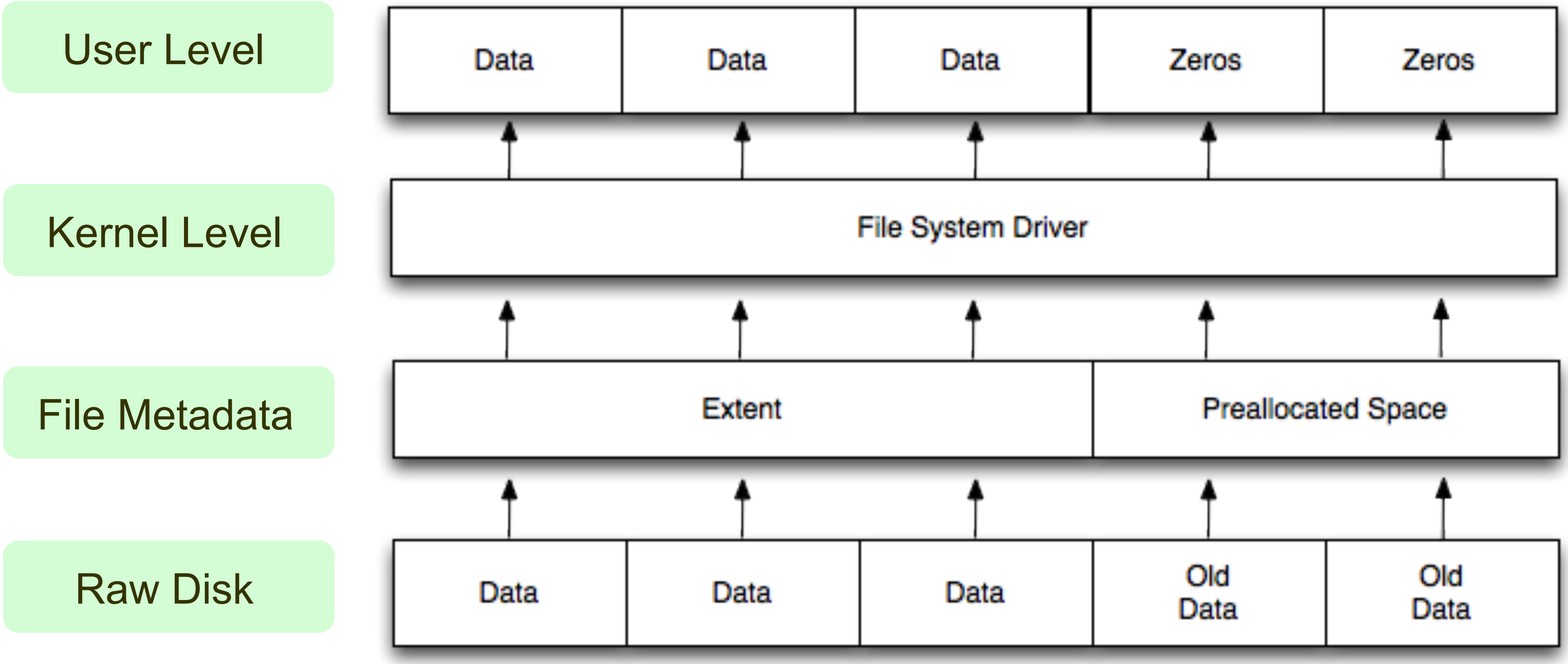
- **Zeroing of inode resident extents depends upon creation of extent tree**
- **Extent index and leaf nodes mixed in with file data blocks**

Forensic Implications II

- **Data**
  - Content may exist in preallocated extents
  - Partially located in H-tree nodes
  - Group Descriptor Growth Blocks
- **Journal**
  - Extend index node journaled

Ext4

What lies beneath?



User Level: Data, Data, Data, Zeros, Zeros

Kernel Level: File System Driver

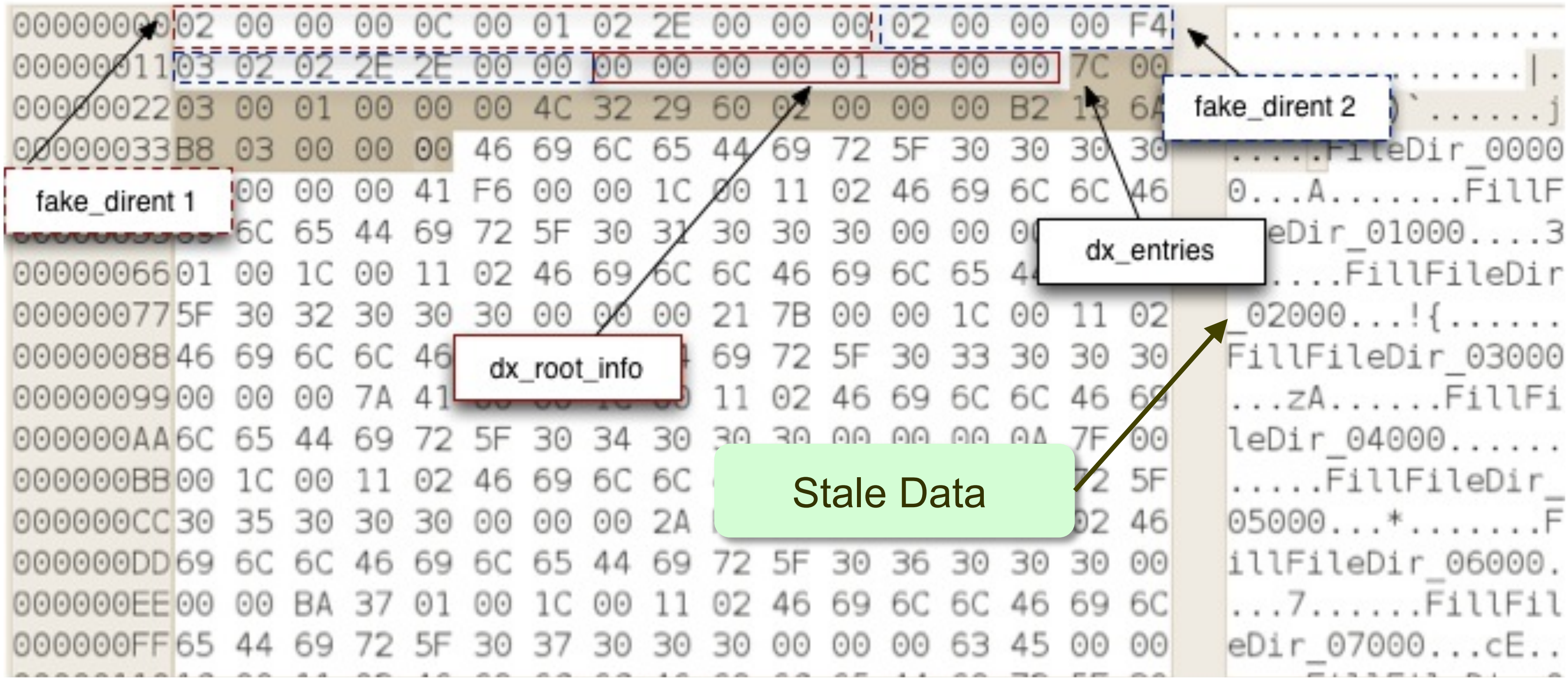
File Metadata: Extent, Preallocated Space

Raw Disk: Data, Data, Data, Old Data, Old Data

Forensic Implications

Stale Data: Directory Indexing

- **Root Node Example**

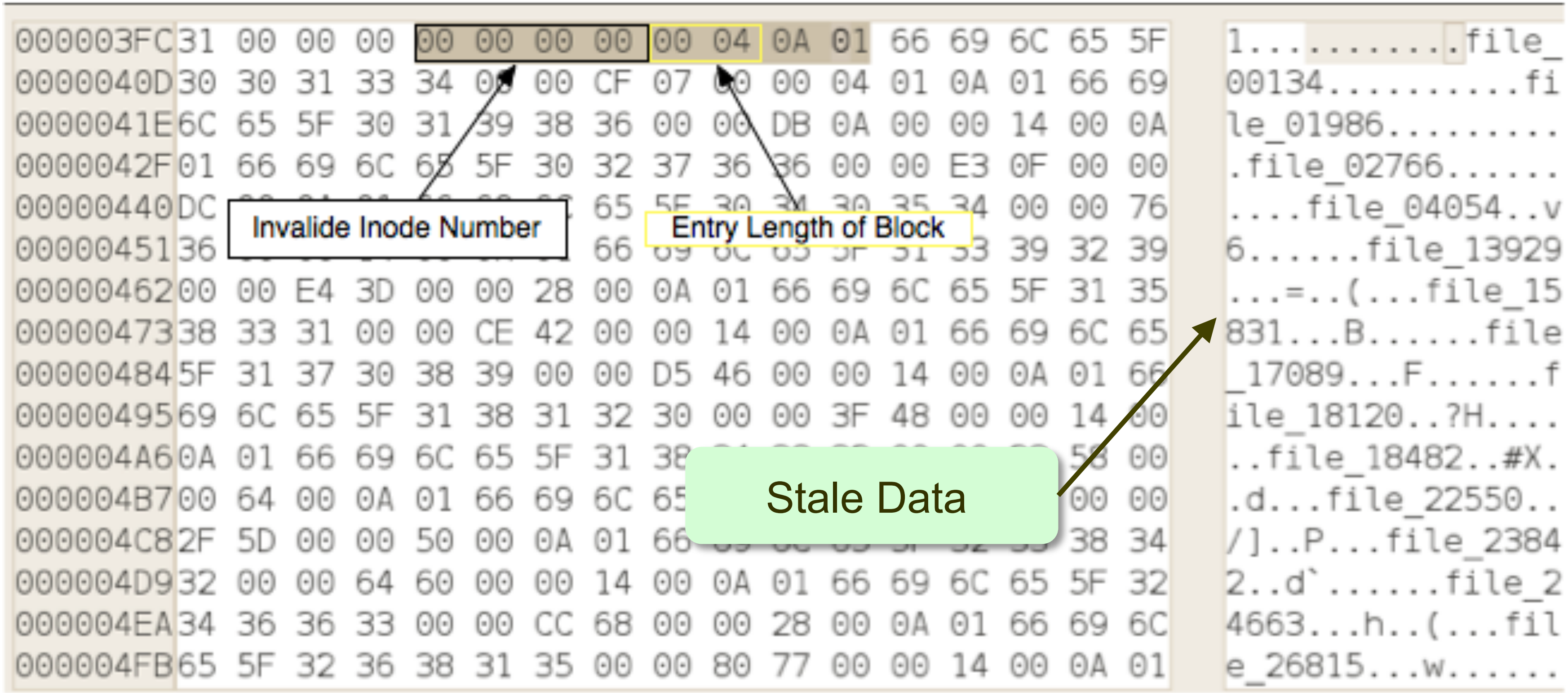


```
0000000002 00 00 00 0C 00 01 02 2E 00 00 00 02 00 00 00 F4   .................
0000001103 02 02 2E 2E 00 00 00 00 00 00 01 08 00 00 7C 00            ......|.
0000002203 00 01 00 00 00 4C 32 29 60 02 00 00 00 B2 1B 6A            )`......j
00000033B8 03 00 00 00 46 69 6C 65 44 69 72 5F 30 30 30 30   .....FileDir_0000
           00 00 00 41 F6 00 00 1C 00 11 02 46 69 6C 6C 46   0...A........FillF
           6C 65 44 69 72 5F 30 31 30 30 30 00 00 00           eDir_01000....3
0000006601 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44          ....FillFileDir
000000775F 30 32 30 30 30 00 00 00 21 7B 00 00 1C 00 11 02   _02000...!{......
0000008846 69 6C 6C 46             69 72 5F 30 33 30 30 30   FillFileDir_03000
0000009900 00 00 7A 41             11 02 46 69 6C 6C 46 69   ...zA......FillFi
000000AA6C 65 44 69 72 5F 30 34 30 30 30 00 00 00 0A 7F 00   leDir_04000......
000000BB00 1C 00 11 02 46 69 6C 6C                   72 5F   .....FillFileDir_
000000CC30 35 30 30 30 00 00 00 2A                   02 46   05000...*.......F
000000DD69 6C 6C 46 69 6C 65 44 69 72 5F 30 36 30 30 30 00   illFileDir_06000.
000000EE00 00 BA 37 01 00 1C 00 11 02 46 69 6C 6C 46 69 6C   ...7......FillFil
000000FF65 44 69 72 5F 30 37 30 30 30 00 00 00 63 45 00 00   eDir_07000...cE..
```

Forensic Implications

Stale Data: Directory Indexing

- **Index Node Example**



```
000003FC31 00 00 00 00 00 00 00 00 04 0A 01 66 69 6C 65 5F   1...........file_
0000040D30 30 31 33 34 00 00 CF 07 00 00 04 01 0A 01 66 69   00134.........fi
0000041E6C 65 5F 30 31 39 38 36 00 00 DB 0A 00 00 14 00 0A   le_01986.........
0000042F01 66 69 6C 65 5F 30 32 37 36 36 00 00 E3 0F 00 00   .file_02766......
00000440DC                   65 5F 30 34 30 35 34 00 00 76   ....file_04054..v
0000045136                   66                39 32 39   6.......file_13929
0000046200 00 E4 3D 00 00 28 00 0A 01 66 69 6C 65 5F 31 35   ...=..(...file_15
0000047338 33 31 00 00 CE 42 00 00 14 00 0A 01 66 69 6C 65   831...B......file
000004845F 31 37 30 38 39 00 00 D5 46 00 00 14 00 0A 01 66   _17089...F......f
0000049569 6C 65 5F 31 38 31 32 30 00 00 3F 48 00 00 14 00   ile_18120..?H....
000004A60A 01 66 69 6C 65 5F 31 38                   58 00   ..file_18482..#X.
000004B700 64 00 0A 01 66 69 6C 65                   00 00   .d...file_22550..
000004C82F 5D 00 00 50 00 0A 01 66                   38 34   /]..P...file_2384
000004D932 00 00 64 60 00 00 14 00 0A 01 66 69 6C 65 5F 32   2..d`......file_2
000004EA34 36 36 33 00 00 CC 68 00 00 28 00 0A 01 66 69 6C   4663...h..(...fil
000004FB65 5F 32 36 38 31 35 00 00 80 77 00 00 14 00 0A 01   e_26815...w......
```

Invalide Inode Number

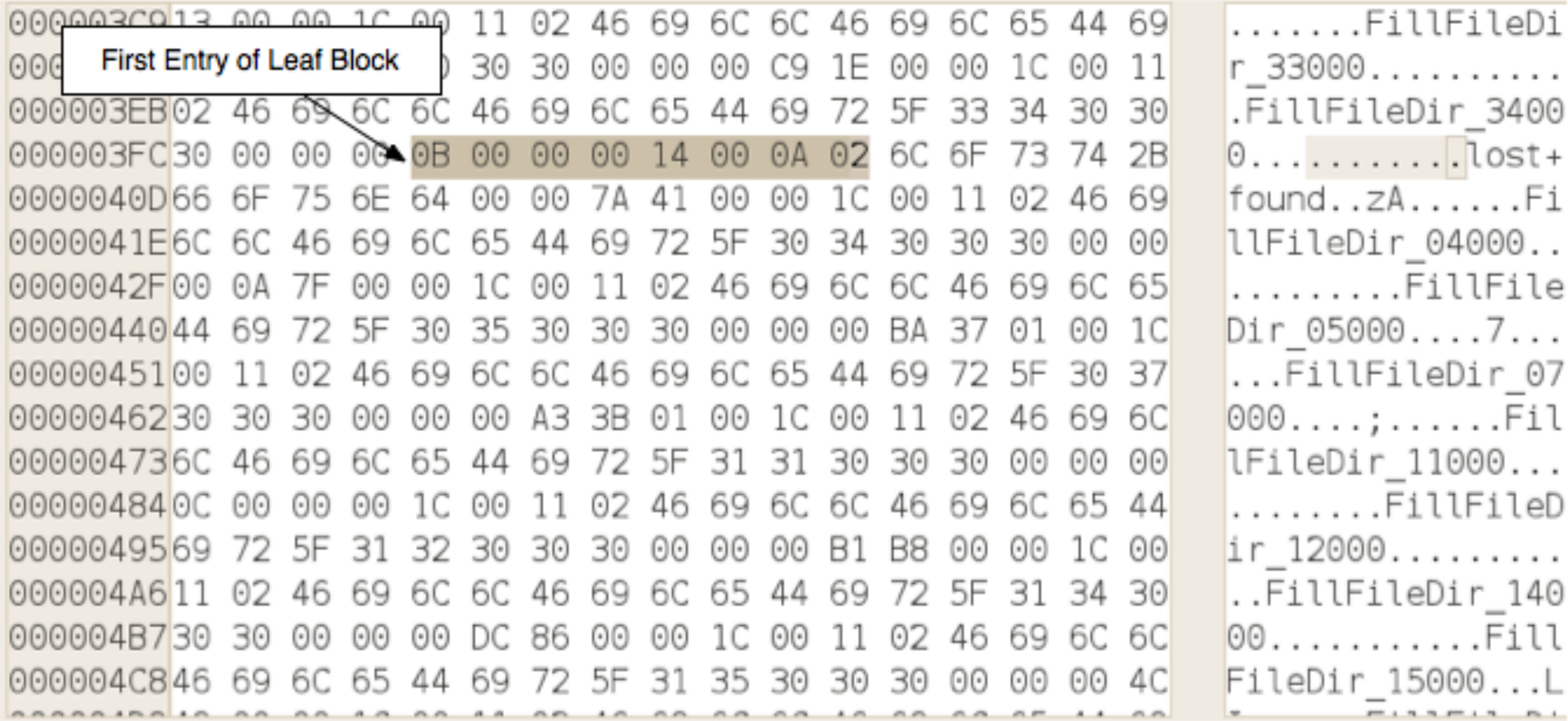
Entry Length of Block

Stale Data

Forensic Implications

Stale Data: Directory Indexing

- **Leaf Node Example**



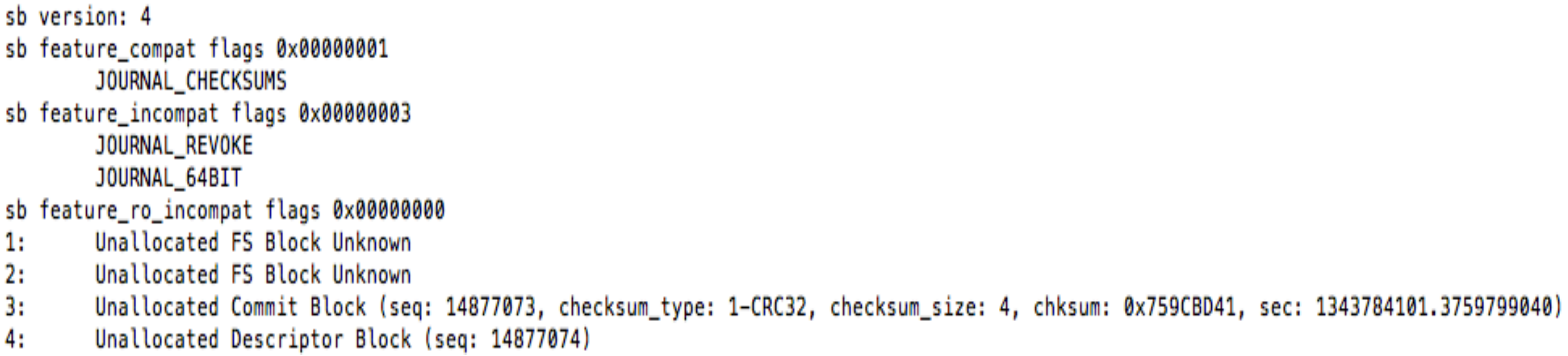
```
000003C913 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44 69   ........FillFileDi
000                    30 30 00 00 00 C9 1E 00 00 1C 00 11   r_33000..........
000003EB02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 33 34 30 30   .FillFileDir_3400
000003FC30 00 00 00 0B 00 00 00 14 00 0A 02 6C 6F 73 74 2B   0...........lost+
0000040D66 6F 75 6E 64 00 00 7A 41 00 00 1C 00 11 02 46 69   found..zA......Fi
0000041E6C 6C 46 69 6C 65 44 69 72 5F 30 34 30 30 30 00 00   llFileDir_04000..
0000042F00 0A 7F 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65   .........FillFile
0000044044 69 72 5F 30 35 30 30 30 00 00 00 BA 37 01 00 1C   Dir_05000....7...
0000045100 11 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 30 37   ...FillFileDir_07
0000046230 30 30 00 00 00 A3 3B 01 00 1C 00 11 02 46 69 6C   000....;......Fil
000004736C 46 69 6C 65 44 69 72 5F 31 31 30 30 30 00 00 00   lFileDir_11000...
000004840C 00 00 00 1C 00 11 02 46 69 6C 6C 46 69 6C 65 44   ........FillFileD
0000049569 72 5F 31 32 30 30 30 00 00 00 B1 B8 00 00 1C 00   ir_12000.........
000004A611 02 46 69 6C 6C 46 69 6C 65 44 69 72 5F 31 34 30   ..FillFileDir_140
000004B730 30 00 00 00 DC 86 00 00 1C 00 11 02 46 69 6C 6C   00...........Fill
000004C846 69 6C 65 44 69 72 5F 31 35 30 30 30 00 00 00 4C   FileDir_15000...L
```

Current Status & Futurework

- **Completed**
  - In depth study of Ext4
- **In Progress**
  - TSK + Ext4
  - https://github.com/kfairbanks/sleuthkit (for now)
- **Next Steps**
  - Ext4 Snapshots
  - Volatility of data
    - H-tree Spaces
    - Group Descriptor Growth Blocks
    - Extent Trees
    - Online Defragmentation
  - Data Hiding Using Journal
    - Do CRCs prevent techniques for hiding data?

TSK + Ext4

jls snapshot

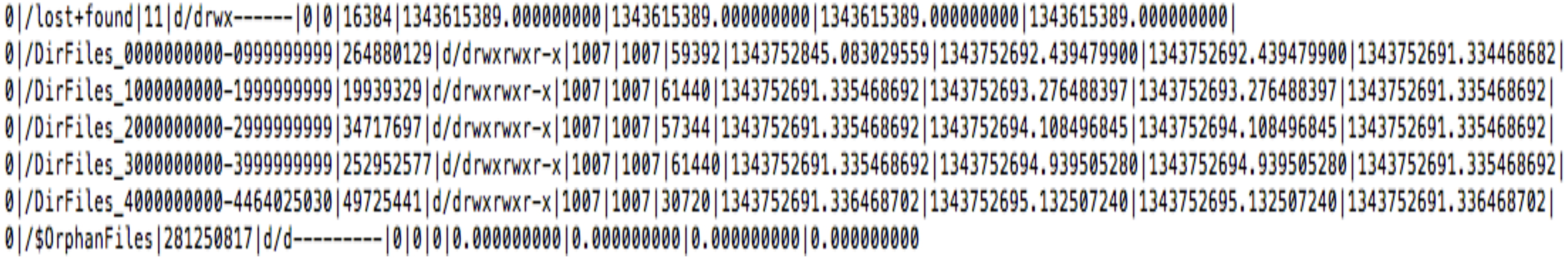


```
sb version: 4
sb feature_compat flags 0x00000001
        JOURNAL_CHECKSUMS
sb feature_incompat flags 0x00000003
        JOURNAL_REVOKE
        JOURNAL_64BIT
sb feature_ro_incompat flags 0x00000000
1:      Unallocated FS Block Unknown
2:      Unallocated FS Block Unknown
3:      Unallocated Commit Block (seq: 14877073, checksum_type: 1-CRC32, checksum_size: 4, chksum: 0x759CBD41, sec: 1343784101.3759799040)
4:      Unallocated Descriptor Block (seq: 14877074)
```

- **Reporting subsecond timestamps**
- **Checksum Type**
- **Checksum value**

TSK + Ext4

fls output

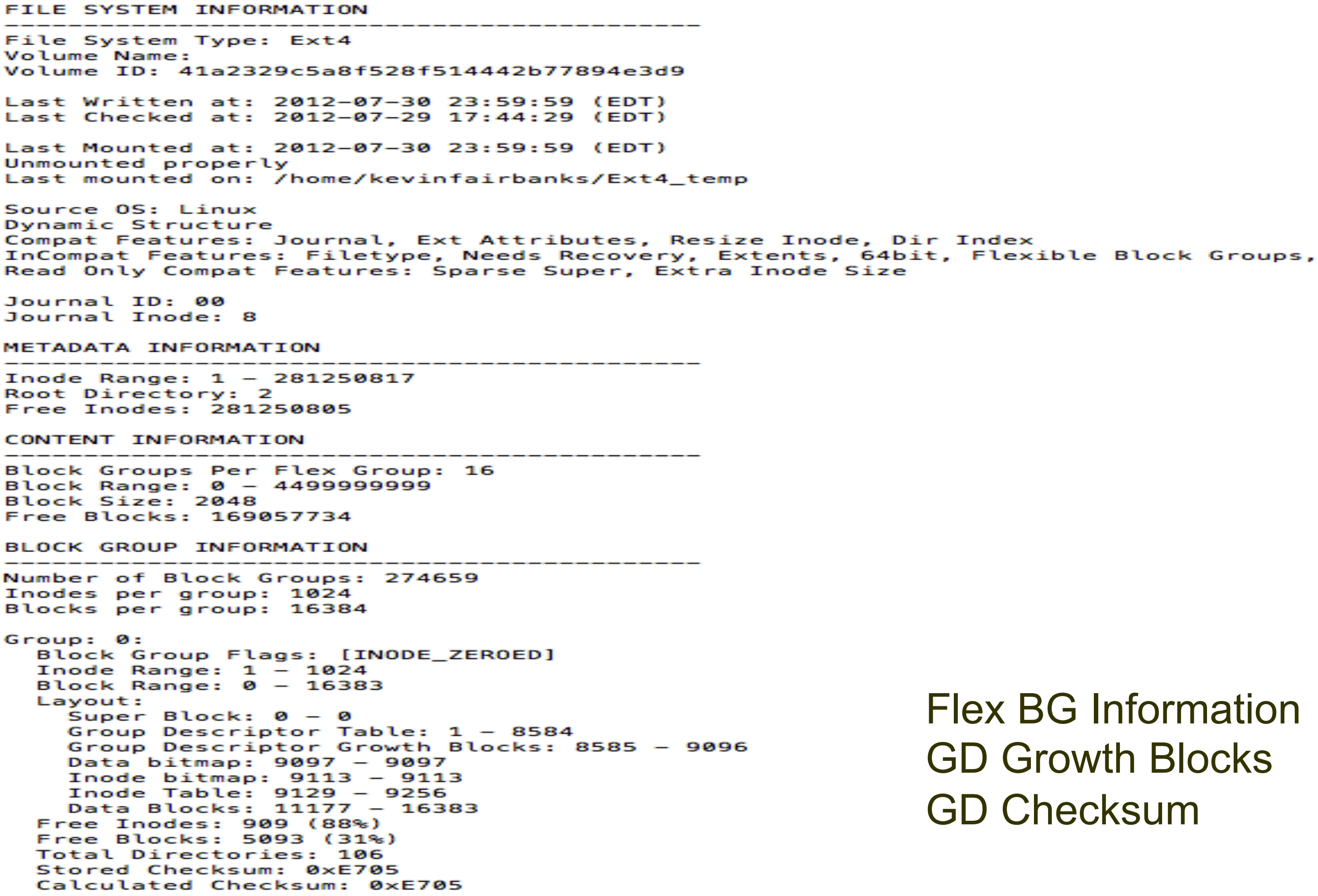


```
0|/lost+found|11|d/drwx------|0|0|16384|1343615389.000000000|1343615389.000000000|1343615389.000000000|1343615389.000000000|
0|/DirFiles_0000000000-0999999999|264880129|d/drwxrwxr-x|1007|1007|59392|1343752845.083029559|1343752692.439479900|1343752692.439479900|1343752691.334468682|
0|/DirFiles_1000000000-1999999999|19939329|d/drwxrwxr-x|1007|1007|61440|1343752691.335468692|1343752693.276488397|1343752693.276488397|1343752691.335468692|
0|/DirFiles_2000000000-2999999999|34717697|d/drwxrwxr-x|1007|1007|57344|1343752691.335468692|1343752694.108496845|1343752694.108496845|1343752691.335468692|
0|/DirFiles_3000000000-3999999999|252952577|d/drwxrwxr-x|1007|1007|61440|1343752691.335468692|1343752694.939505280|1343752694.939505280|1343752691.335468692|
0|/DirFiles_4000000000-4464025030|49725441|d/drwxrwxr-x|1007|1007|30720|1343752691.336468702|1343752695.132507240|1343752695.132507240|1343752691.336468702|
0|/$OrphanFiles|281250817|d/d---------|0|0|0|0.000000000|0.000000000|0.000000000|0.000000000
```

- **Nanosecond timestamps**
- **Creation timestamp**

TSK + Ext4

fsstat output



```
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: Ext4
Volume Name:
Volume ID: 41a2329c5a8f528f514442b77894e3d9

Last Written at: 2012-07-30 23:59:59 (EDT)
Last Checked at: 2012-07-29 17:44:29 (EDT)

Last Mounted at: 2012-07-30 23:59:59 (EDT)
Unmounted properly
Last mounted on: /home/kevinfairbanks/Ext4_temp

Source OS: Linux
Dynamic Structure
Compat Features: Journal, Ext Attributes, Resize Inode, Dir Index
InCompat Features: Filetype, Needs Recovery, Extents, 64bit, Flexible Block Groups,
Read Only Compat Features: Sparse Super, Extra Inode Size

Journal ID: 00
Journal Inode: 8

METADATA INFORMATION
--------------------------------------------
Inode Range: 1 - 281250817
Root Directory: 2
Free Inodes: 281250805

CONTENT INFORMATION
--------------------------------------------
Block Groups Per Flex Group: 16
Block Range: 0 - 4499999999
Block Size: 2048
Free Blocks: 169057734

BLOCK GROUP INFORMATION
--------------------------------------------
Number of Block Groups: 274659
Inodes per group: 1024
Blocks per group: 16384

Group: 0:
  Block Group Flags: [INODE_ZEROED]
  Inode Range: 1 - 1024
  Block Range: 0 - 16383
  Layout:
    Super Block: 0 - 0
    Group Descriptor Table: 1 - 8584
    Group Descriptor Growth Blocks: 8585 - 9096
    Data bitmap: 9097 - 9097
    Inode bitmap: 9113 - 9113
    Inode Table: 9129 - 9256
    Data Blocks: 11177 - 16383
  Free Inodes: 909 (88%)
  Free Blocks: 5093 (31%)
  Total Directories: 106
  Stored Checksum: 0xE705
  Calculated Checksum: 0xE705
```

Flex BG Information
GD Growth Blocks
GD Checksum