TUXERA

NTFS 3G

NTFS for Embedded systems

Jean-Pierre André

Oct 16, 2009

TUXERA NTFS 3G

Needs for Windows storage interoperability

- Network-attached Storage devices (NAS)
- Home entertainment devices
  - Set-top boxed, IPTV, DTV, audio/video players and recorders, game consoles
- Quick-boot solutions: Linux based systems on BIOS
- System-on-a-chip (SoC) solutions, with a dedicated OS
- Demanding expert systems
  - medical, forensics, intelligence, military, data collectors, etc.
  - Data stored locally and processed later on Windows or Unix

Why NTFS ?

- Mass storage needs an interoperable file system
  - Storage plugged elsewhere for further processing
  - Storage devices are being formated as NTFS from factory
- Solution has been FAT32, but it is phasing out
  - 4 GB max file size (no DVD rips)
  - very inefficient for large storage (no bitmap, btree, etc.)
  - Currently, 32GB SD cards are available
  - Limited set, and fixed-size attributes

NTFS main Features

- Storage capacity up to 2^{46}GB
- Efficient storage layout
  - Bitmap for allocator, Btree for indexes
  - Reduced fragmentation and seek times
- Metadata for common Oses
  - OS2, Win32 and Unix by design
- Compression, encryption, sparse files
- Transactional

Design

- HPFS designed ca 1985 for OS2 by IBM and Microsoft
- Design reused by Microsoft with apparent input from VMS
- V1.x from NT 3.1 mid 1993
  - V1.2 : compression, named stream, ACLs
- V3.0 in Win2000
  - Journaling, quota, encryption, sparse files, reparse points
- V3.1 in WinXP up to Vista
  - Symbolic links, transactional NTFS

Alternate offers

- Three generations of GPLed development on Linux
  - Original, led by Martin von Löwis (first release 1997)
  - Linux-ntfs, led by Anton Altaparmakov (first release 2002)
  - ntfs-3g, led by Szabolcs Szakacsits (first release 2007)
- Commercial
  - Paragon
  - Tuxera
- Misc
  - Captive
- Most offers ported to other platforms, and embedded systems

TUXERA

NTFS 3G

I - NTFS

- Files
- Attributes
- Names
- Streams
- Times
- Indexes
- Transactions
- etc.

Structure of a file

- MFT record + variable size attributes
- Standard information (times, version, etc.)
- Names of file
- Streams
- Indexes
- Ownership and permissions
- Metadata stored the same way as ordinary files
  - MFT, bitmap, boot, bad sectors, etc.

File names

- Up to 255 chars (ISO 10646 UTF16LE : any mix of languages)
- Name spaces
  - Win32 and DOS : case insensitive, a few forbidden chars
  - Posix : case sensitive
- A special file defines the lower to upper casing
  - May be redefined to support a non-standard alphabet
- Hard links and short names
- A few reserved file names in the root directory
  - Their first character is a '$'

TUXERA

NTFS 3G

Data streams

- Unnamed data stream : the usual contents
- Named data streams (ADS)
  - External properties of the file (author, summary, ...)
  - Extended attributes for user or system
- Several storage modes for streams
  - Within the root of the file (small files)
  - Attached, plain
  - Sparse
  - Compressed
  - Encrypted

Indexes

- For directories and special files (eg ACL)
- Btree
  - Minimizes the number of seeks
- Several collation schemes
- Ability to process case-sensitive and case-insensitive keys in the same index

Compression

- Transparent to the user or application
- LZ77 algorithm (akin to zip)
- Moderate compression level (about one half for plain text)
- Holes not stored (sparse files)
- Compression acts independently on fixed size blocks (4096 bytes) stored by groups of 16 uncompressed clusters
- Access to any location without decompressing from the beginning
- Most useful for slow devices and limited space

Reparse points

- Attribute defining some processing for accessing a file/dir
- Volume junctions
  - Pointer to a volume defined as a device address
- Directory junctions
  - Pointer to a directory defined as a device address to the directory
  - Evaluated on the server
- Symbolic links (since Vista)
  - Pointer to a directory or file
  - Absolute or relative
  - Evaluated on the client

Encryption

- Transparent to the user, after authentication
- Must be applied to all streams of a file
- Two parts
  - The data encrypted with a symmetric key
  - The key encrypted with public keys of each allowed reader

Access Control (ACL)

- An ACL for each file defines which users are allowed to access the file, and how
- Each user or group is identified by a SID
  - eg S-1-5-21-3141592653-589793238-462643383-1008
- Each access mode is identified by a flag
  - eg read, append, execute, read extended attributes
- Several ACE to be processed in sequence
  - An ACL is a list of ACEs
  - Each ACE is a (user, mode) pair
- ACLs may be defined as inherited from parent directory

Transactions (aka journalling)

- Ability to keep storage integrity despite common errors
  - Power failures
  - Unclean unmounting (pulling the USB cord)
- Each user action is processed as a transactional unit
  - Creating a file, a directory, changing the protections, etc.
- For each internal action an undo/redo record is stored
  - New file : update directory, update index, update bitmap, ...
  - The transaction is committed when the set of user actions is recorded
- When mounting, the unfinished transactions are un/redone
- Transactions do not protect against hardware failures

Ability to recover damaged data

- Most sensitive data are duplicated
  - Boot, MFT, ACLs
- Unwanted deletion of files
  - The root inode may be identified and revalidated
  - The cluster allocation table is specific per file
- Unwanted formatting
  - Most inodes and allocation tables may be recovered
- Unwanted repartitioning
  - The original partition beginning has to be restored
- Numerous commercial recovery programs

TUXERA

NTFS 3G

II - NTFS-3G, currently

- General
- Compression
- Reparse points
- Encryption
- Access control
- Performances

Architecture

- NTFS-3G is an open implementation of NTFS
- It generally relies on FUSE
  - FUSE : an OS plugin to hijack file system calls to a user-mode file system
- Can be compiled for 32-bit or 64-bit OSes
- Has support for both endiannesses
- Ported to numerous operating systems
  - (Linux), MacOS X, FreeBSD, Solaris, etc. (through FUSE or like)
  - Embedded systems (FUSE or direct into the OS)
- Is interoperable with Windows

Posix conformant

- Common functions : read, write, directory list, etc
- UTF-8 file names
- Hard links, symbolic links, pipes, devices, etc.
- Times : modification, access, attribute change
- Extended attributes
- Ownership and permissions

Compression

- Full support for compressed file reading
- File writing
  - Creating and appending to file supported
  - Sparse compressed file supported
  - Overwriting not supported

Reparse points

- Junction points and symlinks made to appear as symbolic links
- Physical addresses and device letter not directly usable
- Target first searched as a mount file parameter
- If not defined, target searched in current volume
  - Search with case insensitivity
  - Translated for use with case sensitivity

Encryption

- Support for backup/restore without decrypting
- No support for encrypting/decrypting

Access control

- The ACLs are approximated to Posix rights
  - rwx for owner, group, other
  - Posix ACL supported
- The users and groups have to be mapped explicitly
  - SID to uid/gid table
  - Default pattern
- The rights to new files are set according to umask or Posix type inheritance
- Static Windows-like inheritance on option
- Special access flags emulated (sticky, setuid, setgid)

Internal NTFS data

- Ability to set data not reachable by Posix-type functions
- File attributes (readonly, hidden, system, archive, etc.)
- Reparse data
- NTFS ACLs
- DOS names
- Times (100ns resolution, incl creation time)
- EFS Info (table of encrypted encryption keys)

Transactions

- Not yet, not perceived as a priority so far
  - Repairing is enough for most users, more reliably than on FAT
- OS+hardware have to guarantee sequence points (“barriers”)
  - No new writes until all previous writes have been secured
  - Reordering of writes between sequence points allowed
  - Not possible with all controllers
  - Not possible currently on Linux
- Unclean unmount is detected at next mount
  - Advised to run an integrity fixer

Footprint and performance

- 24KB code per mount
- 250KB shared library
- 350KB heap (per mount)
- 8KB stack

TUXERA

NTFS 3G

Licensing

- GPL v2
- Released as a standard package in most Linux distributions
  - Fedora, Ubuntu, Suse, Debian, etc.
- Developers have signed an agreement with Tuxera for dual licensing

TUXERA

NTFS 3G

III - Tuxera NTFS

- 100% native read/write kernel driver
- Based on NTFS-3G, developed by same people
- Optimized for high-performance and embedded use
- Low memory & CPU footprint
- Multi-threaded, reentrant, SMP-safe
- Highly portable, wide CPU and kernel version support

Tuxera ?

- Open source developers
  - who reverse engineered NTFS
  - and are developing and maintaining the Linux kernel driver, ntfsprogs, the Linux-NTFS and NTFS-3G open source projects.
- Management experienced in technology & intellectual property law
  - open source, licensing, patents, anti-competition, OEM, embedded sale.

TUXERA NTFS 3G

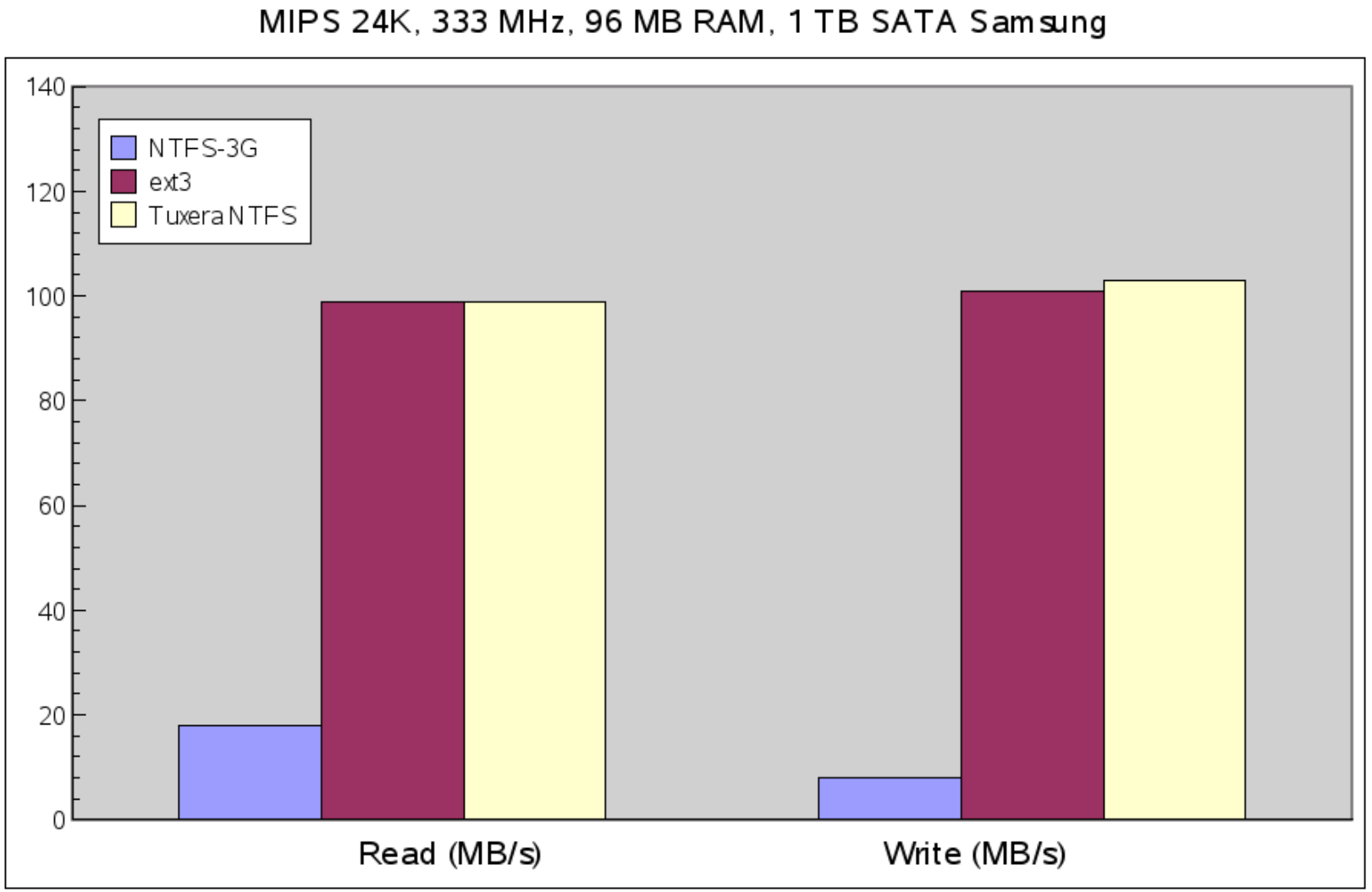
Tuxera NTFS performance

- Comparable or better than popular Linux file systems
- Zero-copy, direct IO
  - Avoiding internal copies and cacheing
- Efficient large and small block size support
- Workload specific operation modes
  - Streaming mode : no cacheing of large files
  - Normal mode : cacheing, except O_DIRECT

TUXERA

NTFS 3G

Performance benchmark



TUXERA

NTFS 3G

Tuxera licensing

- Open-core, dual-licensing
  - like MySQL, Berkeley DB, QT, Asterix, etc
- Proprietary licensing, and compatible with working on open source
  - Open source is a key to software quality

Legal considerations

- Tuxera signed an exFAT intellectual property agreement with Microsoft (Aug 2009)
  - exFAT is another file system promoted by Microsoft
- Tuxera signed an interoperability alliance agreement with Microsoft (Aug 2009)
  - Vendors alliance to share information on interoperability
- Still working with Microsoft on clarifying NTFS legal status
  - One year of discussions now
  - No known NTFS IP violation in Tuxera code
- Tuxera is actively participating and financially supporting legal entities to fight for Free Software and against anti-competition

More...

- NTFS-3G
  - http://www.ntfs-3g.org
- Advanced NTFS-3G
  - http://pagesperso-orange.fr/b.andre
- Tuxera
  - http://www.tuxera.com
  - http://www.tuxera.com/forum