Standard database page header

Every page in a database has a 16 byte standard page header. Various page types have an additional header that follows on from the standard one. The C code representation of the standard header is:

```
struct pag
{
    SCHAR pag_type;
    UCHAR pag_flags;
    USHORT pag_checksum;
    ULONG pag_generation;
    ULONG pag_scn;
    ULONG reserved;
};
```

Pag_type: One byte, signed. Byte 0x00 on the page. This byte defines the page type for the page. Valid page types are:

- 0x00 - Undefined page. You should never see this in a database.
- 0x01 - The database header page. Only ever seen on the very first page of the database, or, on the first page of each database file in a multi-file database.
- 0x02 - The Page Inventory Page (PIP). This page keeps track of allocated and free pages using a bitmap where a 1 means the page is free and a zero shows a used page. There may be more than one PIP in a database, but the first PIP is always page 1.
- 0x03 - Transaction Inventory Page (TIP). A page that keeps track of the stat of transactions. Each transaction is represented by a pair of bits in a bitmap. Valid values in these two bits are:
    - 00 - this transaction is active.
    - 01 - this transaction is in limbo.
    - 10 - this transaction is dead.
    - 11 - this transaction has committed.
- 0x04 - Pointer Page. Each table has one or more of these and this page type keeps track of all the pages that make up the table. Pointer pages are owned by one and only one table, there is no sharing allowed. Each pointer in the array on these pages holds the page number for a type 5 page holding data for the table.
- 0x05 - Data Page. These pages store the actual data for a table.
- 0x06 - Index Root Page. Similar to a type 4 Pointer Page, but applies to indexes only.
- 0x07 - Index B-Tree Page. Similar to the type 5 Data Page, but applies to indexes only.
- 0x08 - Blob Page. Blobs have their own storage within the database. Very large blobs will require a sequence of pages and the type 8 page holds blob data.
- 0x09 - Generator Page. Holds an array of 64 bit generators.
- 0x0a - Page 2 of any database is a Write Ahead Log page. These pages are no longer used. The page will remain blank (filled with binary zero) as it is never used. This page has a standard header like all others.

Pag_flags: One byte, unsigned. Byte 0x01 on the page. This byte holds various flags for the page.

Pag_checksum: Two bytes, unsigned. Bytes 0x02 - 0x03. Checksum for the whole page. No longer used, always 12345, 0x3039. Databases using ODS 8 on Windows NT do have a valid checksum here.

Note: Discussions are underway on the development mailing list on reusing this field as a page number rather than a checksum. From Firebird 3.0, it is possible that this field in the page header will probably have a new name and function.

Pag_generation: Four bytes, unsigned. Bytes 0x04 - 0x07. The page generation number. Increments each time the page is written back to disc.

Pag_scn: Four bytes, unsigned. Bytes 0x08 - 0x0b. Originally used as the sequence number in the Write Ahead Log, but WAL is no longer used. The field was converted to be the SCN number to avoid an ODS change and is now used by nbackup.

Pag_reserved: Four bytes, unsigned. Bytes 0x0c - 0x0f. Reserved for future use. It was originally used for the offset of a page's entry in the Write Ahead Log (WAL), but this is no longer in use.