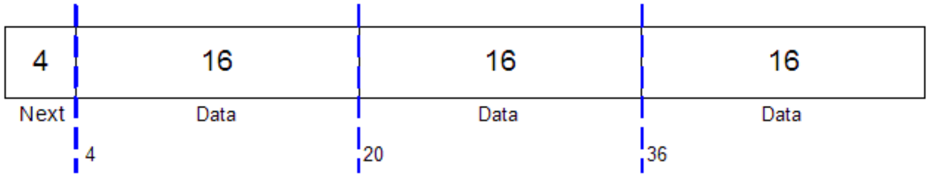
Example 1: 16-byte data, no padding, no header blocks, no alignment.

Field	Size
Next pointer	4 bytes
Padding	0 bytes (not used)
Header block	0 bytes (not used)
Data	16 bytes
Alignment	0/0 bytes (not used)
Page size	52 bytes



Memory dump (wrapped at 32 bytes):

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA B4 3D 33 00 AA AA AA AA AA AA AA AA
AA AA AA AA C4 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
```

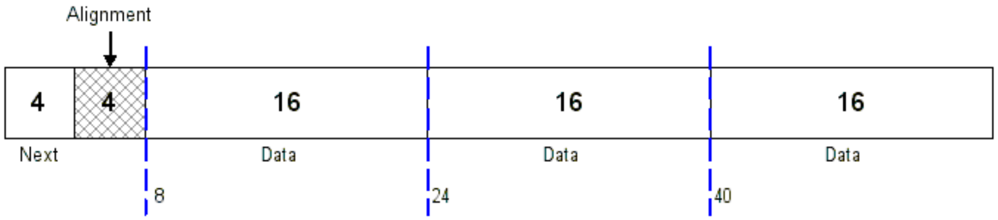
Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 00 00 00 00 AA AA AA AA AA AA AA AA
AA AA AA AA B4 3D 33 00 AA AA AA AA AA AA AA AA
AA AA AA AA C4 3D 33 00 AA AA AA AA AA AA AA AA
AA AA AA AA
```

Note, however, that the data will naturally be aligned on 4-byte boundaries, due to the size of the data.

Example 2: 16-byte data, no padding, no header blocks, 8-byte alignment.

Field	Size
Next pointer	4 bytes
Padding	0 bytes (not used)
Header block	0 bytes (not used)
Data	16 bytes
Alignment	4/0 bytes (left/interblock)
Page size	56 bytes



Memory dump (wrapped at 32 bytes):

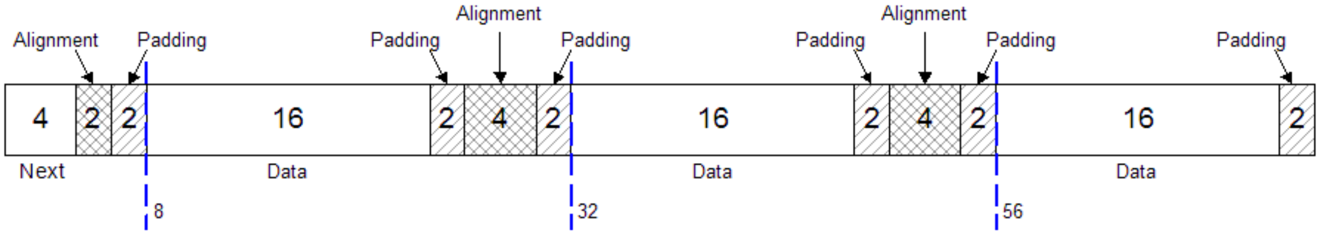
```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 EE EE EE EE 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA B8 3D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA C8 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 EE EE EE EE 00 00 00 00 AA AA AA AA
AA AA AA AA AA AA AA AA B8 3D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA C8 3D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA
```

Example 3: 16-byte data, 2-byte padding (left/right), no header blocks, 8-byte alignment.

Field	Size
Next pointer	4 bytes
Padding	2 bytes
Header block	0 bytes (not used)
Data	16 bytes
Alignment	2/4 bytes (left/interblock)
Page size	74 bytes



Memory dump (wrapped at 32 bytes):

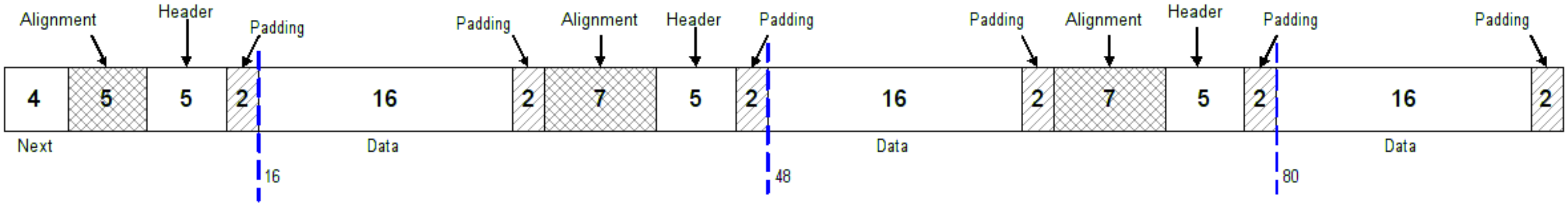
```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 EE EE DD DD 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD EE EE EE EE DD DD
B8 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD EE EE EE EE DD DD D0 3D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA DD DD
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 EE EE DD DD 00 00 00 00 AA AA AA AA
AA AA AA AA AA AA AA AA DD DD EE EE EE EE DD DD
B8 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD EE EE EE EE DD DD D0 3D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA DD DD
```

Example 4: 16-byte data, 2-byte padding (left/right), basic header blocks (5 bytes), 16-byte alignment.

Field	Size
Next pointer	4 bytes
Padding	2 bytes
Basic header block	5 bytes
Data	16 bytes
Alignment	5/7 bytes (left/interblock)
Page size	98 bytes



Memory dump (wrapped at 32 bytes):

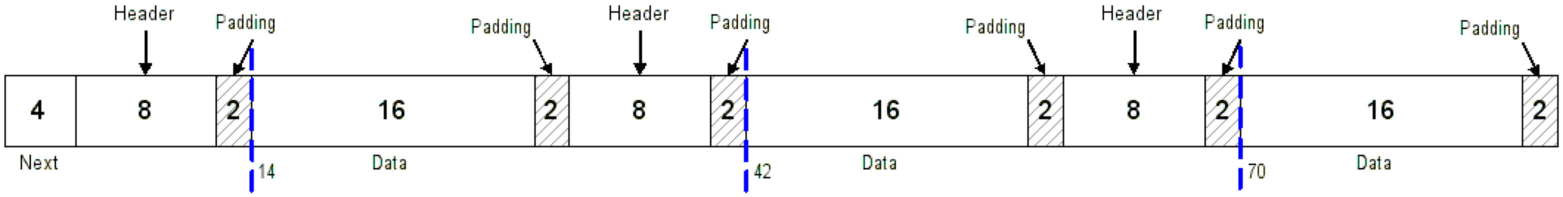
```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 EE EE EE EE EE 00 00 00 00 00 DD DD 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD EE EE EE EE EE EE EE 00 00 00 00 00 DD DD C0 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD EE EE EE EE EE EE EE 00 00 00 00 00 DD DD E0 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 EE EE EE EE EE 00 00 00 00 00 DD DD
00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD EE EE EE EE EE EE EE 00 00 00 00 00 DD DD
C0 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD EE EE EE EE EE EE EE 00 00 00 00 00 DD DD
E0 3D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD
```

Example 5: 16-byte data, 2-byte padding (left/right), extended header blocks with 1 additional byte (8 bytes), no alignment.

Field	Size
Next pointer	4 bytes
Padding	2 bytes
Extended header block	8 bytes
Data	16 bytes
Alignment	0/0 no alignment
Page size	88 bytes



Memory dump (wrapped at 32 bytes):

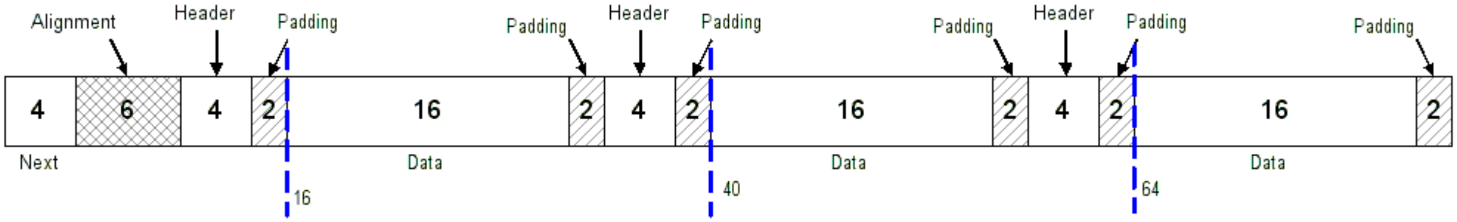
```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 00 00 00 00 00 00 00 00 DD DD 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD
00 00 00 00 00 00 00 00 DD DD 8E 4D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD 00 00 00 00
00 00 00 00 DD DD AA 4D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 00 00 00 00 00 00 00 00 DD DD 00 00
00 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD
00 00 00 00 00 00 00 00 DD DD 8E 4D 33 00 AA AA
AA AA AA AA AA AA AA AA AA AA DD DD 00 00 00 00
00 00 00 00 DD DD AA 4D 33 00 AA AA AA AA AA AA
AA AA AA AA AA AA DD DD
```

Example 6: 16-byte data, 2-byte padding (left/right), external header blocks (4 bytes, 32-bit), 8-byte alignment.

Field	**Size**
Next pointer	4 bytes
Padding	2 bytes
External header block	4 bytes
Data	16 bytes
Alignment	6/0 alignment
Page size	82 bytes



Memory dump (wrapped at 32 bytes):

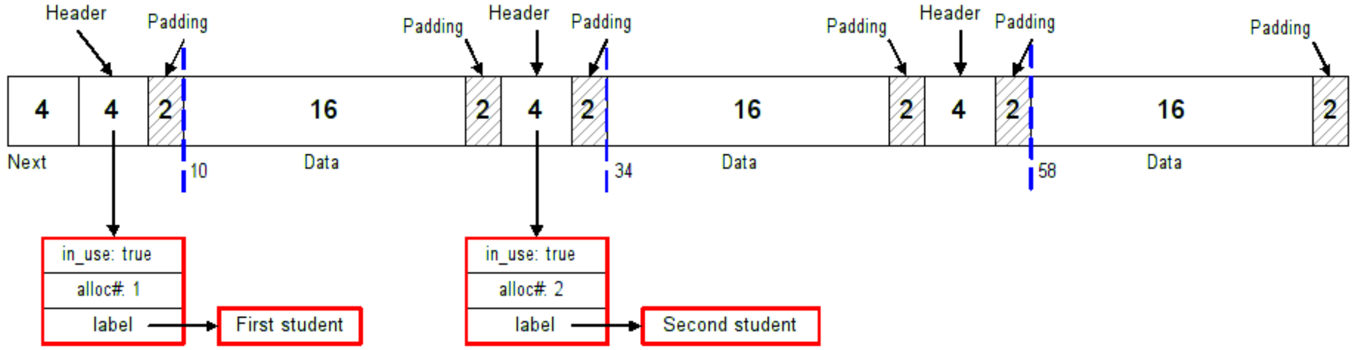
```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 EE EE EE EE EE EE 00 00 00 00 DD DD 00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD 00 00 00 00 DD DD 90 4D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD 00 00 00 00 DD DD
A8 4D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA DD DD
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 EE EE EE EE EE EE 00 00 00 00 DD DD
00 00 00 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD 00 00 00 00 DD DD 90 4D 33 00 AA AA AA AA
AA AA AA AA AA AA AA AA DD DD 00 00 00 00 DD DD
A8 4D 33 00 AA AA AA AA AA AA AA AA AA AA AA AA
DD DD
```

Example 7: 16-byte data, 2-byte padding (left/right), external header blocks (4 bytes, 32-bit) showing the dynamically-allocated structs and dynamically-allocated strings, no alignment, assuming 32-bit computer.

Field	Size
Next pointer	4 bytes
Padding	2 bytes
External header block	4 bytes
Data	16 bytes
Alignment	no alignment
Page size	76 bytes



Memory dump (wrapped at 32 bytes):

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
00 00 00 00 40 4E 33 00 DD DD BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB DD DD B8 4E 33 00
DD DD BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB BB DD DD 00 00 00 00 DD DD CC CC CC CC CC CC
CC CC CC CC CC CC CC CC CC CC DD DD
```

Wrapped at 16 bytes:

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
00 00 00 00 40 4E 33 00 DD DD BB BB BB BB BB BB
BB BB BB BB BB BB BB BB BB BB DD DD B8 4E 33 00
DD DD BB BB BB BB BB BB BB BB BB BB BB BB BB BB
BB BB DD DD 00 00 00 00 DD DD CC CC CC CC CC CC
CC CC CC CC CC CC CC CC CC CC DD DD
```

The diagram above is the result of allocating all 3 blocks, then freeing the right-most block. The signatures reflect this fact.