
Subject: Re: BufferPainter::Clear() optimization
Posted by [mirek](#) on Mon, 18 May 2020 11:43:39 GMT
[View Forum Message](#) <> [Reply to Message](#)

This variation of basically the same thing seems a tiny bit faster:

```
void Fill3a(RGBA *t, RGBA c, int len)
{
    __m128i val4 = _mm_set1_epi32(*(int*)&c);
    auto Set4 = [&](int at) { _mm_storeu_si128((__m128i*)(t + at), val4); };
    auto Set4S = [&](int at) { _mm_stream_si128((__m128i*)(t + at), val4); };
    if(len >= 32) {
        if(len > 1024*1024 / 16 && ((uintptr_t)t & 3) == 0) { // for really huge data, bypass the cache
            while((uintptr_t)t & 15) { // align to 16 bytes for SSE
                *t++ = c;
                len--;
            }
            do {
                Set4S(0); Set4S(4); Set4S(8); Set4S(12);
                Set4S(16); Set4S(20); Set4S(24); Set4S(28);
                t += 32;
                len -= 32;
            }
            while(len >= 32);
            _mm_sfence();
        }
        else
            do {
                Set4(0); Set4(4); Set4(8); Set4(12);
                Set4(16); Set4(20); Set4(24); Set4(28);
                t += 32;
                len -= 32;
            }
            while(len >= 32);
    }
    if(len & 16) {
        Set4(0); Set4(4); Set4(8); Set4(12);
        t += 16;
    }
    if(len & 8) {
        Set4(0); Set4(4);
        t += 8;
    }
    if(len & 4) {
        Set4(0);
        t += 4;
    }
}
```

```
if(len & 2) {  
    t[0] = t[1] = c;  
    t += 2;  
}  
if(len & 1)  
    t[0] = c;  
}
```
