Subject: Re: BufferPainter::Clear() optimization
Posted by mirek on Thu, 21 May 2020 11:28:28 GMT
View Forum Message <> Reply to Message

OK, so I could not stop digging and found last important ingredient: alignment matters!

```
void FillX(void *p, dword data, int len)
{
 dword *t = (dword *)p;
 if(len < 4) {
  if(len & 2) {
   t[0] = t[1] = t[len - 1] = data;
   return;
  }
  if(len & 1)
   t[0] = data;
  return;
 }

 __m128i val4 = _mm_set1_epi32(data);
 auto Set4 = [&](int at) { _mm_storeu_si128((__m128i *)(t + at), val4); };

 Set4(len - 4); // fill tail
 if(len >= 16) {
  Set4(0); // align up on 16 bytes boundary
  const dword *e = t + len;
  t = (dword *)(((uintptr_t)t | 15) + 1);
  len = e - t;
  e -= 16;
  if(len >= 1024*1024) { // for really huge data, bypass the cache
   huge_memsetd(t, data, len);
   return;
  }
  while(t <= e) {
   Set4(0); Set4(4); Set4(8); Set4(12);
   t += 16;
  }
 }
 if(len & 8) {
  Set4(0); Set4(4);
  t += 8;
 }
 if(len & 4)
  Set4(0);
}
```

This is about twice as fast as Fill7a for len > 60 (up to cache bypass limit).