

C++ Club Meeting Notes

Gleb Dolgich

2018-12-06

San Diego trip reports

- ▶ Botond Ballo
- ▶ JeanHeyd Meneide

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (1/13)

Video



CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (2/13)

The slide features a timeline from 1980 to 1984, highlighting various milestones in 3D graphics:

- 1980:** MAGI
- 1981:** ABEL, III, DIGITAL EFFECTS
- 1982:** CRANSTON CSURI, SideFX (red dot), POLYGON PICTURES
- 1983:** RETROSPECTIVE (with a TRON car image), DIGITAL PROD., OMNIBUS
- 1984:** SideFX logo

A large orange circle marks the year 1982. The slide is titled "Historical Roots" and includes the text "Patterns and Techniques Used in the Houdini 3D Graphics Application". The CppCon 2018 logo is in the top right.

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (3/13)

The image is a composite of two screenshots. On the left is a photo of the boy band New Kids on the Block from 1982, featuring five young men in front of a red background. On the right is a video thumbnail for CppCon 2018. The thumbnail features a man in a black polo shirt standing and gesturing with his hands. The text "INTRODUCTION" is at the top, "MARK ELENDT" is in the center, and "Patterns and Techniques Used in the Houdini 3D Graphics Application" is at the bottom. The SideFX logo is visible in the bottom right corner of the thumbnail. The overall background is dark blue.

1982

INTRODUCTION

SideFX

cppcon | 2018
THE C++ CONFERENCE • BELLEVUE, WASHINGTON

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (4/13)

cppcon | 2018

THE C++ CONFERENCE • BELLEVUE, WASHINGTON

Early Days

RETROSPECTIVE



MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (5/13)

The slide is a composite image. On the left, there is a timeline showing two portraits: Greg Hermanovic (1985) and Kim Davidson (1986). Below the portraits is a screenshot of the SideFX software interface titled "RETROSPECTIVE". The interface includes a color palette, a node graph with nodes like "PRISMS" and "Side Effects", and a small thumbnail of a book cover for "Side Effects". A red dot on the timeline marks the year 1987. On the right side of the slide, there is a photo of Mark Elendt standing at a podium, speaking. The podium has a nameplate that reads "MARK ELENDT". Below the photo, there is a title card with the text "Patterns and Techniques Used in the Houdini 3D Graphics Application". The top right corner of the slide features the CppCon 2018 logo.

Retrospective

Greg Hermanovic Kim Davidson

1985 1986 1987 1988 1989

RETROSPECTIVE

PRISMS

Side Effects

SideFX

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

cppcon | 2018
THE C++ CONFERENCE • BELLEVUE, WASHINGTON

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (6/13)

The image is a collage of two photographs. On the left, a group of five young men, identified as the SideFX team from 1997, are posed together against a blue background. They are dressed in casual 90s-style clothing, including a baseball cap, sunglasses, and striped shirts. A small circular logo with a swirl is in the top left corner, and the year '1997' is in the top right. The word 'RETROSPECTIVE' is centered above the photo. The SideFX logo is in the bottom right corner. On the right, a photo of Mark Elendt, now an older man with grey hair, is shown from the waist up, wearing a black polo shirt with the SideFX logo. Below his photo is the name 'MARK ELENDT'. To the right of his photo is a block of text: 'Patterns and Techniques Used in the Houdini 3D Graphics Application'. At the bottom right is the website 'CppCon.org'. The overall background is dark with faint circular patterns.

1997

RETROSPECTIVE

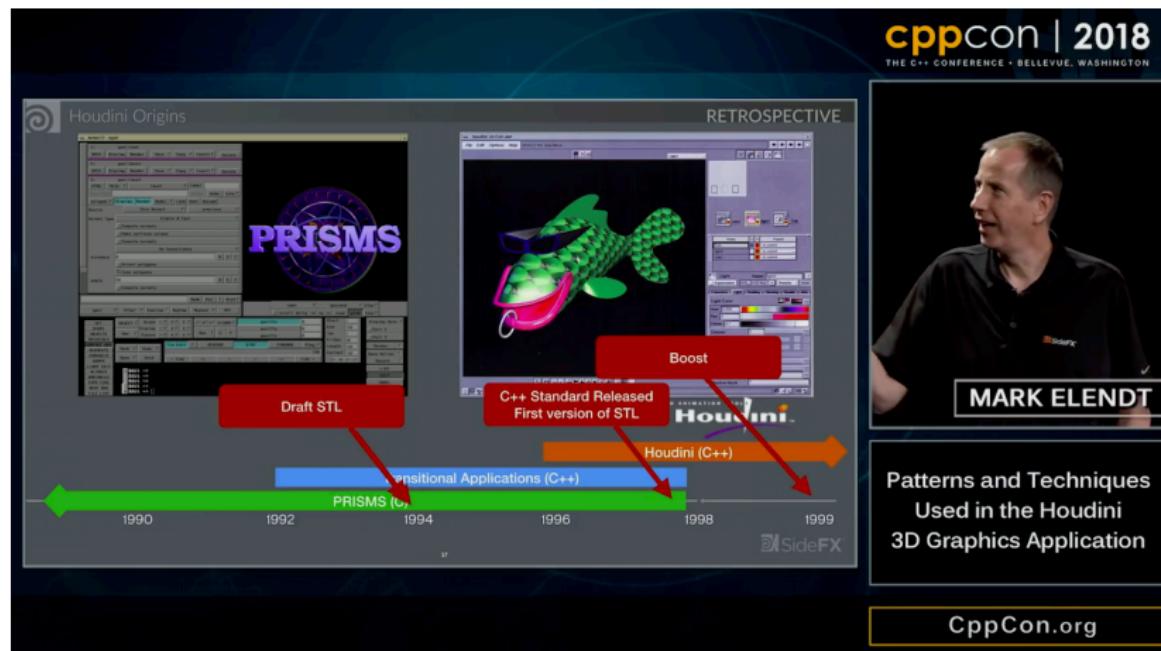
SideFX

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (7/13)



CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (8/13)

The image shows a screenshot of a video player interface. In the top right corner, the CppCon 2018 logo is visible, with the text "THE C++ CONFERENCE • BELLEVUE, WASHINGTON". The main video frame on the right shows a man with glasses, identified as Mark Elendt, speaking at a podium. He is wearing a dark polo shirt and has a microphone attached to his shirt. A silver laptop is open on the podium. Below the video frame, the name "MARK ELENDT" is displayed in a white box. To the left of the video frame, there is a large slide with the "SideFX" logo at the bottom right. The slide has a dark background and contains the following text:

- VFX Reference Platform
- 2015
 - gcc v4.8.2
 - boost 1.5.x
 - Qt 4.8.x
 - Python 2.7.x
- SideFX

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (9/13)

VFX Reference Platform

2016

SOFTWARE ECOSYSTEM

- gcc v4.8.2 -> unchanged
- boost 1.5.x -> boost1.5.8
- Qt 4.8.x -> Qt 5.6.1
- Python 2.7.x -> Python 2.7.5
- **C++11**

SideFX

MARK ELENDT

**Patterns and Techniques
Used in the Houdini
3D Graphics Application**

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (10/13)

The image shows a screenshot of a video player interface. In the top right corner, the CppCon 2018 logo is visible, featuring the text "cppcon | 2018" and "THE C++ CONFERENCE • BELLEVUE, WASHINGTON". The main content area displays a presentation slide. On the left, there's a logo for "VFX Reference Platform" and the text "SOFTWARE ECOSYSTEM". The central part of the slide has a dark background with white text. At the top, it says "2018". Below that, under "VFX Reference Platform", there's a bulleted list of software updates: "gcc v4.8.2 -> gcc v6.3.1", "boost 1.5.8 -> boost 1.6.1", "Qt 5.6.1 -> unchanged", "Python 2.7.5 -> unchanged", and "**C++11 -> C++14**". At the bottom right of the slide, the SideFX logo is present. To the right of the slide, a video frame shows a man with short hair, wearing a black polo shirt with a small logo on the chest, speaking. Below the video frame, the name "MARK ELENDT" is displayed in a white box. Further down, another white box contains the title of the presentation: "Patterns and Techniques Used in the Houdini 3D Graphics Application". At the very bottom right of the video player, the website "CppCon.org" is shown.

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (11/13)

The slide is from CppCon 2018, featuring a presentation by Mark Elendt. The title of the talk is "Patterns and Techniques Used in the Houdini 3D Graphics Application".

Code Editor Content:

```
template <typename T>
class UT_Array
{
    UT_Array() {}

    void growCapacity(size_t size) {
        if (_array)
            _array = (T *)realloc(_array, size*sizeof(T));
        else
            _array = (T *)malloc(size*sizeof(T));
        _size = size;
    }
}
```

Speaker: MARK ELENDT

Title Card:

Patterns and Techniques
Used in the Houdini
3D Graphics Application

SideFX Logo:

SideFX

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (12/13)

The slide is titled "HOUDINI GEOMETRY". It features two code snippets and a 3D visualization of a triangle.

Data Representation

```
Points = {  
    float3 P[4];  
    float3 Velocity[4];  
    float Temperature[4];  
};  
  
Triangles {  
    vector<point> Vertices[2];  
    string Shader[2];  
};
```

HOUDINI GEOMETRY

```
Points[] = {  
    P = {{0,0,0},{2,0,0},  
          {1,1,0},{-2,-9,0}},  
    Velocity = {{1,0,0},{1,0,0},  
               {1,0,0},{1,0,0}},  
    Temperature = {20,20,  
                  20,20},  
}  
  
Faces[] = {  
    Vertices = {{0,2,3},{0,1,2}},  
    Shader = {"green","green"},  
}
```

SideFX

MARK ELENDT

**Patterns and Techniques
Used in the Houdini
3D Graphics Application**

CppCon.org

CppCon 2018: Mark Elendt “Patterns and Techniques Used in the Houdini 3D Graphics Application” (13/13)

cppcon | 2018
THE C++ CONFERENCE • BELLEVUE, WASHINGTON

Data Representation HOUDINI GEOMETRY

```
template <typename POD_T>
class UT_PageArray {
    class PageData {
        POD_T * _data;
    };

    PageData *_pages;

    POD_T &operator[](size_t i) {
        size_t page, offset;
        splitIndex(i, page, offset);
        return _pages[page]{offset};
    }

    inline void splitIndex(size_t i, size_t &page, size_t &offset)
    {
        page = i >> PAGE_BITS;
        offset = i & PAGE_MASK;
    }
};
```

SideFX

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

Package management is hard, let's mine some Bitcoin (1/2)

 **Kenn White**
@kennwhite

Holy hell, Node. A package with 2 million downloads a week and the maintainer hands over control to a rando stranger? And now it's mining cryptocurrency. Wow.

[github.com/dominictarr/ev...](https://github.com/dominictarr/event-stream)
pic.twitter.com/MkqhHzjc1

dominictarr commented 5 days ago

he emailed me and said he wanted to maintain the module, so I gave it to him. I don't get anything from maintaining this module, and I don't even use it anymore, and hasn't for years.

Like 41 Retweet 49 Like 16 Like 3 Like 11 Like 21

dominictarr commented 5 days ago

Owner

note: I no longer have publish rights to this module on npm.

Like 13 Retweet 11 Like 18 Like 1

1,161 Likes 952 Retweets

26 Nov 2018 at 19:10 via Twitter Web Client

 **Jackson Palmer**
@ummjackson

Yikes. The [@BitPay](#) Copay wallet was/is vulnerable to keys being stolen due to the "event-stream" [@npmjs](#) module containing malware because [@dominictarr](#) handed over maintenance of the module to a random person who emailed him. Millions of other NPM module users also affected. 😱

Kevin Beaumont 😱 @GossiTheDog 20h

NPM library with 2m installs has a backdoor, looks to be some kind of Trojan (stealer?) [github.com/dominictarr/ev...](https://github.com/dominictarr/event-stream)

235 Likes 128 Retweets

26 Nov 2018 at 19:02 via Twitter Web Client

Package management is hard, let's mine some Bitcoin (2/2)

- ▶ Issue on GitHub “I don't know what to say”
- ▶ Details on the npm blog
- ▶ event-stream vulnerability explained
- ▶ Bleeping Computer
- ▶ Exploiting developer infrastructure is insanely easy
- ▶ The Node.js Ecosystem Is Chaotic and Insecure

Who is STL? I mean the person, not the library

Reddit

Hey. I'm Stephan T. Lavavej ("Steh-fin Lah-wah-wade"), and I've worked on MSVC's STL since 2007. I've also worked on several Standard proposals that were accepted (notably the transparent operator functors). I filmed a bunch of videos for MS's Channel 9 years ago, introducing various Core Language and Standard Library topics, and I've given talks at C++Now (formerly BoostCon) and CppCon which have been recorded.

Real world problems with #pragma once?

Reddit

No.

Corentin Jabot:

Pragma once leads to UB if part of the sources are visible from several included paths which are hardlink of each others.

Good. If you do that, you deserve to suffer!

Pointer-to-member-functions can be tricky

- ▶ Post
- ▶ Snippet
- ▶ Raymond Chen: Pointers to member functions are very strange animals

Prepare thy Pitchforks: A De-facto Standard Project Layout

- ▶ Early Reddit post
- ▶ Later Reddit post
- ▶ Blog post
- ▶ GitHub repo

Reimplementing NumPy in C++

- ▶ [NumCpp](#)
- ▶ [xtensor](#)

Other linear algebra libraries

- ▶ [Blaze](#)
- ▶ [Eigen](#)
 - ▶ [the official repo](#)
 - ▶ [docs](#)

Visual C++ Team Blog - std::any: How, when, and why

Post

When you need to store an object of an arbitrary type, pull std::any out of your toolbox. Be aware that there are probably more appropriate tools available when you do know something about the type to be stored.

C++ Best Practices, by Jason Turner

[GitHub](#)

Quote

Andrey Mokhov (@andreymokhov) via Twitter:

*Inside every large program there is a small build system
struggling to get out.*