

C++ Club Meeting Notes

Gleb Dolgich

2018-11-01

Pre-San Diego Papers (1/5)

- ▶ Bartek's Coding Blog: Five Awesome C++ Papers for San Diego
 - ▶ `constexpr std::string` and `constexpr std::vector`
 - ▶ Extended structured bindings
 - ▶ Pattern matching
 - ▶ Text formatting
 - ▶ A concurrent associative data structure with an unsynchronised view, slides

Pre-San Diego Papers (2/5)

- ▶ JeanHeyd Meneide: Simple, Easy Papers
 - ▶ Nested inline namespaces
 - ▶ nodiscard should have a reason
 - ▶ `void main` – *Hmmm... don't think so*
 - ▶ Byteswapping for fun&&nuf
 - ▶ offsetof for the Modern Era
 - ▶ namespace std { namespace fs = filesystem; } – *Meh*
 - ▶ Literal Suffixes for `size_t` and `ptrdiff_t`
- ▶ Reddit

Pre-San Diego Papers (3/5)

- ▶ JeanHeyd Meneide: Optional Choices to Make
 - ▶ Monadic operations for std::optional
 - ▶ A simple, practical optional reference for C++
 - ▶ “The good news is that this paper lets the C++ Committee continue to wage the holy war about how comparison and assignment for such a type should work while providing a way forward for not only std::optional<T&>, but variant<T&, ...> and std::expected<T&>.”
 - ▶ Jonathan Boccaro: Why Optional References Didn’t Make It In C++17
 - ▶ Reddit
- ▶ Reddit

Pre-San Diego Papers (4/5)

- ▶ JeanHeyd Meneide: Paper Review II
 - ▶ Feature Presentation
 - ▶ Forward from `initializer_list`
 - ▶ A view of 0 or 1 elements: `view::maybe`
 - ▶ Proposing `unless`
 - ▶ Immediate functions
- ▶ Reddit

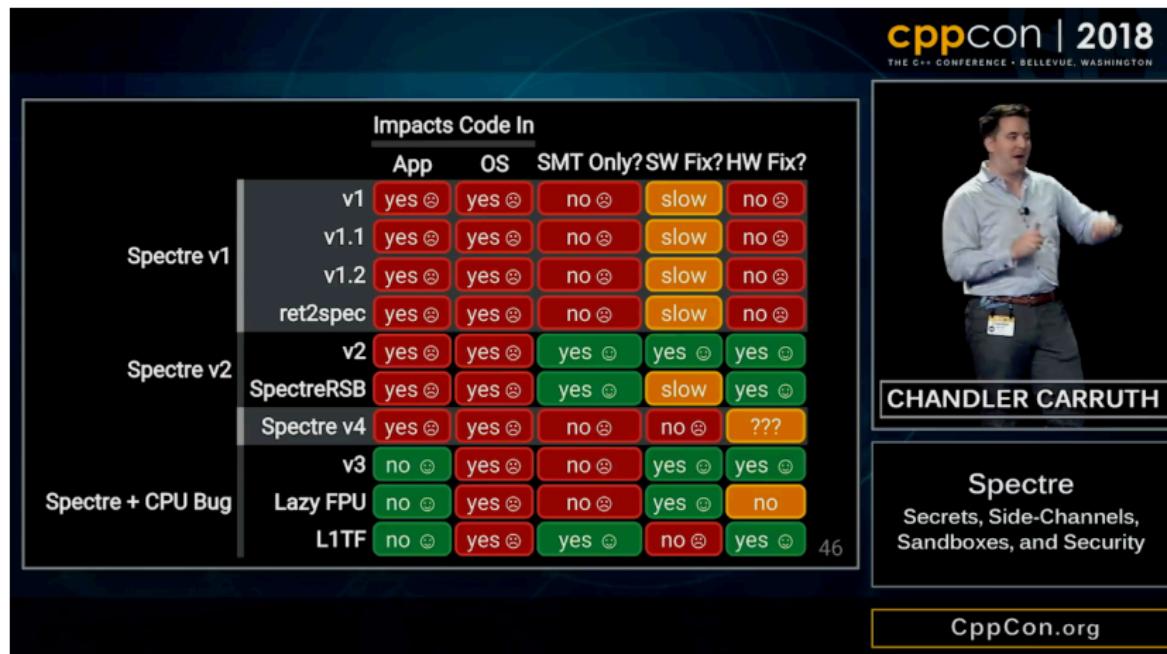
The simplest papers are often the worst for bikeshedding. – Niall Douglas

Pre-San Diego Papers (5/5)

- ▶ `Implicit constexpr`
- ▶ `Package Specification, Notes on C++ Package Management, Package Ecosystem Plan`
- ▶ `Canonical Project Structure`
 - ▶ “If you have to disregard every rule and recommendation in these guidelines but one, for example, because you are working on an existing library, then insist on this: **public header inclusions must use the library name as a directory prefix.**”
- ▶ `find_backward`
- ▶ `flat_map` and `flat_set`
- ▶ `char8_t`: A type for UTF-8 characters and strings

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (1/9)

Video



The slide features a grid titled "Impacts Code In" comparing various Spectre variants and patches across different environments. The columns represent "App", "OS", "SMT Only?", "SW Fix?", and "HW Fix?". The rows include Spectre v1 variants (v1, v1.1, v1.2, ret2spec), Spectre v2 variants (v2, SpectreRSB, Spectre v4), and Spectre + CPU Bug variants (v3, Lazy FPU, L1TF). The "SMT Only?" column uses green for yes and red for no. The "SW Fix?" column uses green for yes and red for no. The "HW Fix?" column uses green for yes and red for no. The "App" and "OS" columns have mixed colors.

		Impacts Code In				
		App	OS	SMT Only?	SW Fix?	HW Fix?
Spectre v1	v1	yes ☺	yes ☺	no ☹	slow	no ☹
	v1.1	yes ☺	yes ☺	no ☹	slow	no ☹
	v1.2	yes ☺	yes ☺	no ☹	slow	no ☹
	ret2spec	yes ☺	yes ☺	no ☹	slow	no ☹
Spectre v2	v2	yes ☺	yes ☺	yes ☺	yes ☺	yes ☺
	SpectreRSB	yes ☺	yes ☺	yes ☺	slow	yes ☺
	Spectre v4	yes ☺	yes ☺	no ☹	no ☹	???
Spectre + CPU Bug	v3	no ☹	yes ☺	no ☹	yes ☺	yes ☺
	Lazy FPU	no ☹	yes ☺	no ☹	yes ☺	no
	L1TF	no ☹	yes ☺	yes ☺	no ☹	yes ☺

46

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

CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (2/9)

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

Retpolines

- Developed by Google
- Requires recompiling source
- Mitigates Spectre v2 (and SpectreRSB in restricted cases)
- May be faster than STIBP for current CPUs.
- Likely slower and less strong than STIBP on future CPUs

<http://bit.ly/2R50i8J-retpoline>, <http://bit.ly/2NOxtPJ-retpoline-intel>

63



CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (3/9)

The screenshot shows a video player interface. At the top, it displays "No Service" with a signal icon, the time "09:41", and the CppCon 2018 logo with battery level at 94%. Below the logo, it shows "-30:48" and "-25:40". The main video frame shows Chandler Carruth speaking on stage. The slide he is presenting has a dark background with three circular icons on the left: a lock, a memory chip, and a gear. The text on the slide reads "Retpolines overhead: under 3% (PGO+ThinLTO)" and includes the number "75" in the bottom right corner. The video player has standard controls at the bottom: a lock icon, back, forward, play/pause, and next. The bottom right corner features the CppCon logo and the text "dts HEADPHONE".

No Service 09:41
X 56:46 -30:48
47:18 25:40

Spectre: Secrets, Side-Channels, Sandboxes, and Security" CppCon - 20181002 - CppCon

Retpolines overhead: under 3%
(PGO+ThinLTO)

75

CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

dts HEADPHONE

CppCon

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (4/9)

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

```
# %bb.1:  
    movq    (%rdi), %rax  
    leaq    8(%rsp), %rsi  
    leaq    24(%rsp), %rdx  
    callq   *48(%rax)  
    movq    24(%rsp), %rdi  
    testq   %rdi, %rdi  
    je     .LBB0_3  
  
# %bb.1:  
    movq    (%rdi), %rax  
    movq    48(%rax), %r11  
    leaq    8(%rsp), %rsi  
    leaq    24(%rsp), %rdx  
    callq   __llvm_retpoline_r11  
    movq    24(%rsp), %rdi  
    testq   %rdi, %rdi  
    je     .LBB0_3
```



CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

65

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (5/9)

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

```
_@_llvm_retpoline_r11:          # @_llvm_retpoline_r11
# %bb.0:
    callq  .LBB1_2
.LBB1_1:                      # Block address taken
                                # => This Inner Loop Header: Depth=1
    pause
    lfence
    jmp   .LBB1_1
    .p2align 4, 0x90
.LBB1_2:                      # Block address taken
    movq  %r11, (%rsp)
    retq
```

74



CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (6/9)

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

Speculative Load Hardening (SLH)

- Automatic hardening against v1
- Developed by Google
- Changes compiler generated code to make v1 attacks impossible
- Very complex: one of the most complicated low-level transforms in LLVM
- Easy to deploy: clang++ -mspeculative-load-hardening ...

<http://bit.ly/2NPcgVY-slh>

77

CHANDLER CARRUTH

Spectre

Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org



CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (7/9)

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

SLH overhead is HUGE: 30% - 40% CPU

90



CHANDLER CARRUTH

**Spectre
Secrets, Side-Channels,
Sandboxes, and Security**

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (8/9)

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

Isolate secret data from risky code

- Sandbox untrusted code (or code handling untrusted input) from data w/ OS/HW barrier (process boundary for example)
- Effective against essentially all known vulnerabilities
- Only realistic mitigation for v4 (SSB)
- Also protects against non-spectre side channel attacks like Heartbleed
- Every browser moving to this model via site- (or origin-) isolation

<http://bit.ly/2zEivmN-chromium-post-spectre>

91



CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org

CppCon 2018: Chandler Carruth “Spectre: Secrets, Side-Channels, Sandboxes, and Security” (9/9)

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

Conclusion

- Spectre: misspeculation + side channel -> leak secrets
- New, active area for research -> ongoing influx of vulnerabilities
- Have a threat model, because we can't afford to mitigate everything
- Tailor mitigations to each application's risk and performance
- Convince CPU vendors to make these problems go away

97



CHANDLER CARRUTH

Spectre
Secrets, Side-Channels,
Sandboxes, and Security

CppCon.org

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

Video

The image displays a video player interface for a CppCon 2018 session. At the top right, the text "cppcon | 2018" and "THE C++ CONFERENCE • BELLEVUE, WASHINGTON" is visible. On the left, a thumbnail for the "Scientific and Technical Academy Awards" shows several people on stage receiving awards. On the right, a larger video frame shows a man in a black polo shirt speaking. Below the video frame, the name "MARK ELENDT" is displayed. In the bottom right corner of the video frame, there is a logo for SideFX. The bottom right corner of the entire interface also contains the text "Patterns and Techniques Used in the Houdini 3D Graphics Application". At the very bottom right, the website "CppCon.org" is shown.

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

INTRODUCTION

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

SideFX

CppCon.org

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

Historical Roots

RETROSPECTIVE

MAGI

ABEL

III

DIGITAL EFFECTS

DIGITAL PROD.

OMNIBUS

CRANSTON CSURI

PDI

POLYGON PICTURES

1980 1981 1982 1983 1984

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” (3/13)

The slide features a collage of images. On the left is a photograph of the British new wave band Duran Duran from their 1982 album 'Rio'. The band members are posed against a red background. On the right is a photograph of Mark Elendt, a man with short grey hair, wearing a black polo shirt with the SideFX logo, standing on a stage and gesturing with his hands. The top right corner of the slide displays the 'cppcon | 2018' logo with the subtitle 'THE C++ CONFERENCE • BELLEVUE, WASHINGTON'. The bottom right corner contains the text 'CppCon.org'.

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)

The slide deck for Mark Elendt's talk at CPPCon 2018 features a dark blue header with the conference logo and text. Below the header, there are four main sections: a collage of images from the 'Early Days' of Houdini, a retrospective photo of a man working at a desk, a video of Mark Elendt speaking, and a summary of the talk's content.

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

Early Days

RETROSPECTIVE

MARK ELENDT

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

CppCon.org

A collage of four images from the 'Early Days' of Houdini: a large green server rack, a tall white server cabinet, a man working at a desk with multiple monitors, and a close-up of a SideFX logo on a circular device.

A photograph of a man sitting at a desk in an office, working on a computer. The desk has multiple monitors and various office equipment.

A video frame showing Mark Elendt, a man with short hair wearing a black polo shirt with a SideFX logo, speaking to an audience.

A text box containing the title of the talk: "Patterns and Techniques Used in the Houdini 3D Graphics Application".

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

Retrospective

RETROSPECTIVE

PRISMS

SideFX

1985 1986 1987 1988 1989

Greg Hermanovic Kim Davidson

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

CppCon.org

The slide is part of a retrospective session at CppCon 2018. It features a timeline from 1985 to 1989, showing the progression of the SideFX logo and the development of the SideFX Houdini software. Portraits of Greg Hermanovic and Kim Davidson are included, along with a screenshot of the Houdini interface. Mark Elendt is shown speaking on stage.

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

1997

RETROSPECTIVE

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” (7/13)

The slide is titled "Houdini Origins" and "RETROSPECTIVE". It features two screenshots of the Houdini interface. The left screenshot shows the "PRISMS" logo and the "Draft STL" button. The right screenshot shows a 3D model of a fish and the "Boost" button. A timeline at the bottom spans from 1990 to 1999. A green arrow points from 1990 to 1994, labeled "PRISMS (C++)". A blue arrow points from 1992 to 1996, labeled "Conventional Applications (C++)". A red arrow points from 1994 to 1999, labeled "Houdini (C++)". A yellow arrow points from 1996 to 1999, labeled "SideFX". The "cppcon | 2018" logo is in the top right corner. On the right side of the slide, there is a portrait of Mark Elendt and text for his talk.

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” (8/13)

VFX Reference Platform

SOFTWARE ECOSYSTEM

2015

- gcc v4.8.2
- boost 1.5.x
- Qt 4.8.x
- Python 2.7.x

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” (9/13)

The image shows a presentation slide from CppCon 2018. At the top right, the logo "cppcon | 2018" is displayed above the text "THE C++ CONFERENCE • BELLEVUE, WASHINGTON". On the left, there is a dark grey rectangular area containing the text "VFX Reference Platform" and "SOFTWARE ECOSYSTEM". In the center, the year "2016" is at the top, followed by a bulleted list of software dependencies: "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", and "C++11". Below this list is the SideFX logo. On the right side of the slide, there is a video frame showing a man, identified as "MARK ELENDT" in a black polo shirt, gesturing with his hands while speaking. At the bottom right of the slide, the text "Patterns and Techniques Used in the Houdini 3D Graphics Application" is displayed, and at the very bottom right, the website "CppCon.org" is shown.

VFX Reference Platform

SOFTWARE ECOSYSTEM

2016

- 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 presentation slide from CppCon 2018. In the top right corner, the text "cppcon | 2018" and "THE C++ CONFERENCE • BELLEVUE, WASHINGTON" is displayed. On the left side, there is a dark rectangular area containing the text "VFX Reference Platform" and "SOFTWARE ECOSYSTEM". In the center, the year "2018" is at the top, followed by a bulleted list of software updates: "gcc v4.8.2 -> gcc v6.3.1", "boost 1.5.8 -> boost1.6.1", "Qt 5.6.1 -> unchanged", "Python 2.7.5 -> unchanged", and "C++11 -> C++14". At the bottom right of this central area is the SideFX logo. On the right side of the slide, there is a video frame showing a man speaking, identified as "MARK ELENDT". Below the video frame, the title of the talk is listed: "Patterns and Techniques Used in the Houdini 3D Graphics Application". At the bottom right of the slide, the website "CppCon.org" is mentioned.

VFX Reference Platform

SOFTWARE ECOSYSTEM

2018

- gcc v4.8.2 -> gcc v6.3.1
- boost 1.5.8 -> boost1.6.1
- Qt 5.6.1 -> unchanged
- Python 2.7.5 -> unchanged
- C++11 -> C++14

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” (11/13)

Array Class

CUSTOM CONTAINERS

```
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;
    }
}
```

std::trivially_relocatable

MARK ELENDT

Patterns and Techniques
Used in the Houdini
3D Graphics Application

SideFX

CppCon.org

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

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

Data Representation HOUDINI GEOMETRY



SideFX

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

```
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"},  
}
```

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

Standard Library Algorithms: Changes and Additions in C++17

VCBlog

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.

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](#)

Library: SQLite ORM

- ▶ [Code](#)

- ▶ Licence: BSD-2-Clause
- ▶ C++14

Library: Inja - a template engine for modern C++

► Code

- Licence: MIT
- Header-only
- Uses NLochmann's [JSON library](#)
- [Conan wrapper](#)

```
1 json data;
2 data["name"] = "world";
3 inja::render("Hello {{ name }}!", data); // Returns "Hello world!"
```

Library: C++ REST SDK (formerly Casablanca) by Microsoft

- ▶ **Code**

- ▶ Licence: MIT
- ▶ C++11
- ▶ Supports Windows, Linux, macOS, iOS, Android

The C++ REST SDK is a Microsoft project for cloud-based client-server communication in native code using a modern asynchronous C++ API design. This project aims to help C++ developers connect to and interact with services.

Library: Caffe2 - A New Lightweight, Modular, and Scalable Deep Learning Framework

- ▶ [Website](#)
- ▶ [Code](#)
 - ▶ Licence: Apache-2.0

Tool: Superluminal profiler for Windows

Website

- ▶ Combines sampling and instrumentation
- ▶ Visualizes thread communication flow
- ▶ Kernel-level callstacks
- ▶ Dynamic filtering of areas of interest
- ▶ High frequency sampling (8 KHz)
- ▶ Timeline view, call graph, source view
- ▶ 7-day free trial, then EUR 99/149/289

Conan, vcpkg or build2?

Reddit

- ▶ Pragmatic choice: vcpkg or Conan (they work today and are complete enough)
- ▶ Pragmatic no-brainer choice: vcpkg (it's the simplest and it have more packages ready)
- ▶ Pragmatic but need finer control choice: Conan (it gives more options)
- ▶ (Very) Long term choice: Build2 (shows great promises because it uses a coherent model...)
- ▶ Ideal choice (from the future): help SG15 (the group reflecting on tools vs C++) define interfaces for build systems and dependency managers so that your choice is not impacted by your dependencies choices.

Improving C++ Builds with Split DWARF

Article

```
1 $ g++ -c -g -fPIC -fPIC main.cpp -o main.o  
2 $ g++ main.o -o app
```

Having some fun with higher-order functions

- ▶ Article by Barry Revzin
- ▶ Boost.HOF

Compile-time raytracer by Tristan Brindle

- ▶ [Code](#)
- ▶ [Reddit](#)

Twitter



Victor Zverovich
@vzverovich

2 Replies

Preferably narrated by David Attenborough.

06/10/2018, 18:42 (Saturday)
Twitter Web Client

4 Likes | 1 Retweet | Thread >

Reply | Retweet | Like | Block



Victor Zverovich @vzverovich 2 days
Is it possible to get the C++ standard as an audiobook? Asking for a friend. #cpp #cplusplus