

# C++ on Sea trip report



©Philip Craig



 **Steve Love**  
@IAmSteveLove

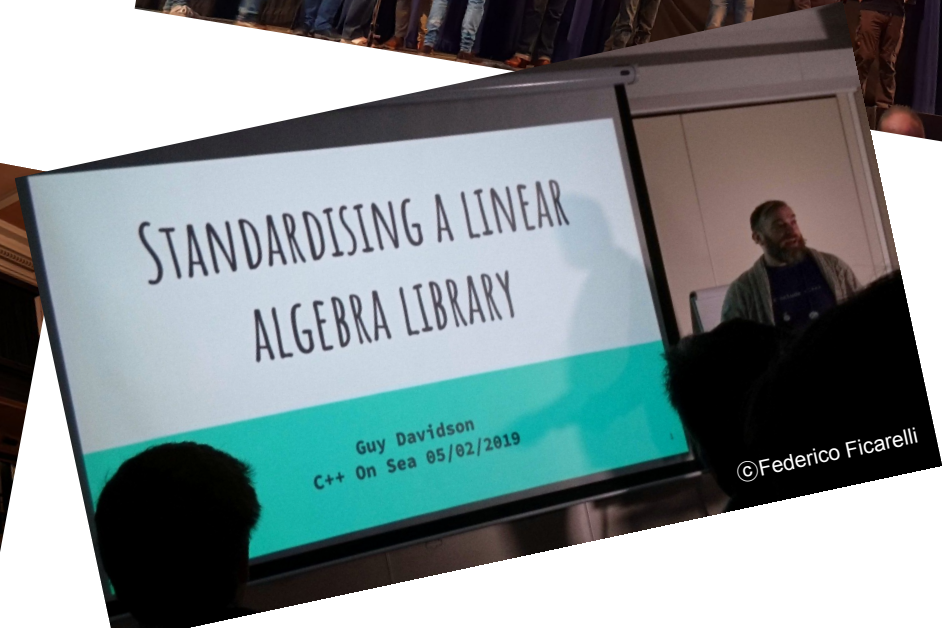


@cpponsea #cpponsea so who was it who chose a British seaside town for a February conf? :->

♡ 7 11:04 AM - Feb 4, 2019

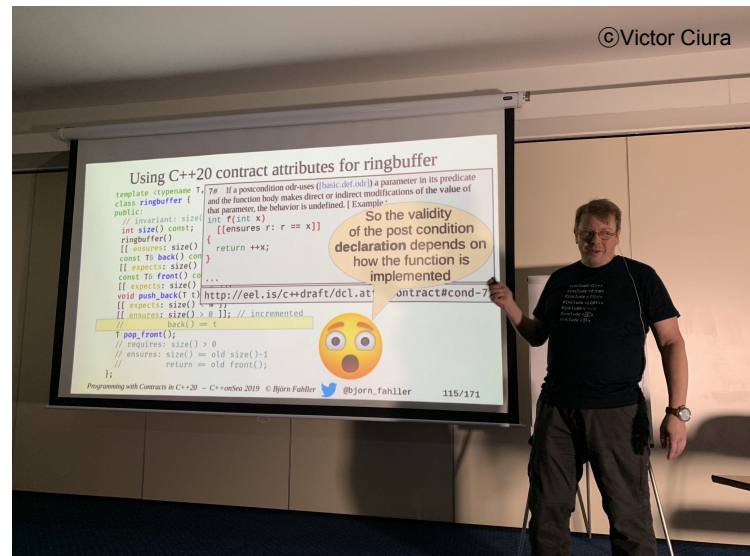
**Folkestone, UK**  
**2 day, 4 tracks**  
**1 workshops day**







©Arne Mertz



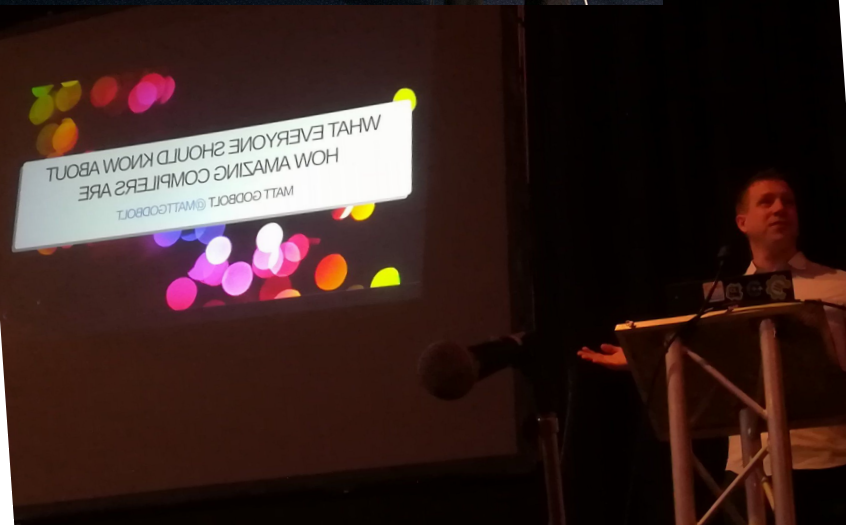
©Victor Ciura

## Initialisation in C++17

Version 2 – Copyright (c) 2019 Timur Doumler

Type var	Default init ; <code>value;</code>	Copy init = <code>value;</code>	Direct init ( <code>args</code> );	Value init ();	Empty braces {}; = {};	Direct list init { <code>args</code> };	Copy list init = { <code>args</code> };
<b>Built-in types</b>	<b>Uninitialised.</b> Variables w/ static storage duration: Zero-initialised	Initialised with value (via conversion sequence)	1 arg: Init with arg >1 arg: <b>Doesn't compile</b>	Zero-initialised	Zero-initialised	1 arg: Init with arg >1 arg: <b>Doesn't compile</b>	1 arg: Init with arg >1 arg: <b>Doesn't compile</b>
<b>auto</b>	<b>Doesn't compile</b>	Initialised with value	Initialised with value	<b>Doesn't compile</b>	<b>Doesn't compile</b>	1 arg: Init with arg >1 arg: <b>Doesn't compile</b>	Object of type <code>std::initializer_list</code>
<b>Aggregates</b>	<b>Uninitialised.</b> Variables w/ static storage duration: Zero-initialised***	<b>Doesn't compile</b>	<b>Doesn't compile</b> (but will in C++20)	Zero-initialised***	Aggregate init**	1 arg: implicit copy/move ctor if possible. Otherwise aggregate init**	1 arg: implicit copy/move ctor if possible. Otherwise aggregate init**
<b>Types with <code>std::initializer_list</code> ctor</b>	Default ctor	Matching ctor (via conversion sequence), explicit ctors not considered	Matching ctor	Default ctor	Default ctor if there is one, otherwise <code>std::initializer_list</code> ctor	<code>std::initializer_list</code> ctor if possible, otherwise matching ctor	<code>std::initializer_list</code> ctor if possible, otherwise matching ctor****
<b>Other types with no user-provided* default ctor</b>	Members are default-initialised	Matching ctor (via conversion sequence), explicit ctors not considered	Matching ctor	Zero-initialised***	Zero-initialised***	Matching ctor	Matching ctor****
<b>Other types</b>	Default ctor	Matching ctor (via conversion sequence), explicit ctors not considered	Matching ctor	Default ctor	Default ctor	Matching ctor	Matching ctor****

\*not user-provided = not user-declared, or user-declared as `=default` inside the class definition  
 \*\*Aggregate init copy-inits all elements with given initialiser, or value-inits them if no initialiser given  
 \*\*\*Zero initialisation zero-initialises all elements and initialises all padding to zero bits  
 \*\*\*\*Copy-list-initialisation considers explicit ctors, too, but doesn't compile if such a ctor is selected



©Gareth Lloyd