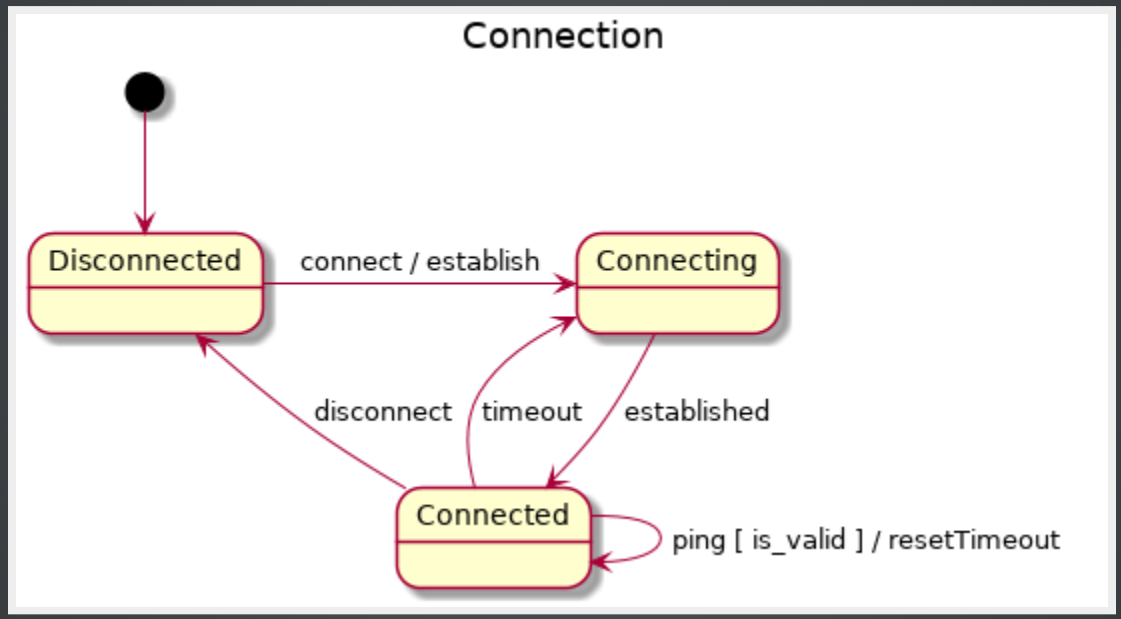
C++ NOW 2017

# [BOOST].SML
# STATE MACHINE LANGUAGE

Kris Jusiak, Quantlab Financial

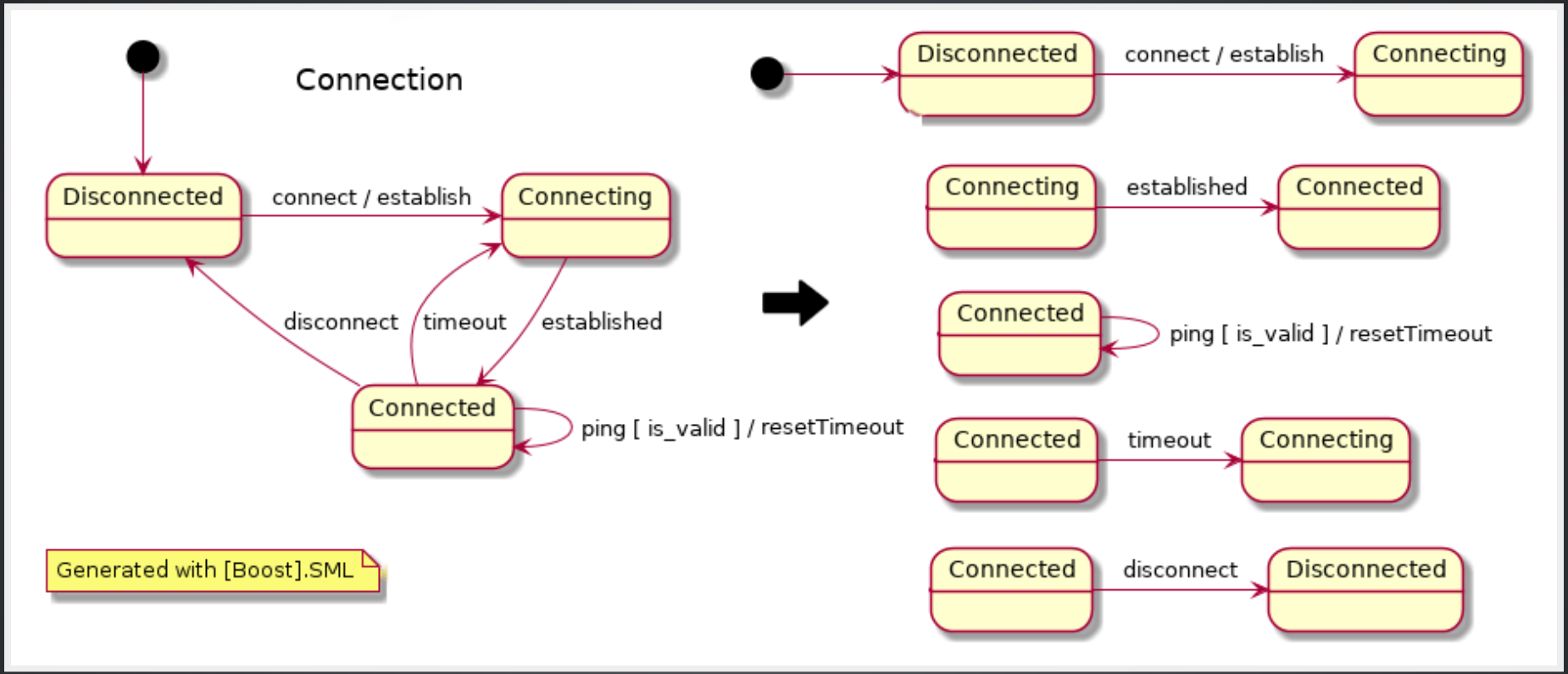
kris@jusiak.net | @krisjusiak | linkedin.com/in/kris-jusiak

# CONNECTION - STATE DIAGRAM

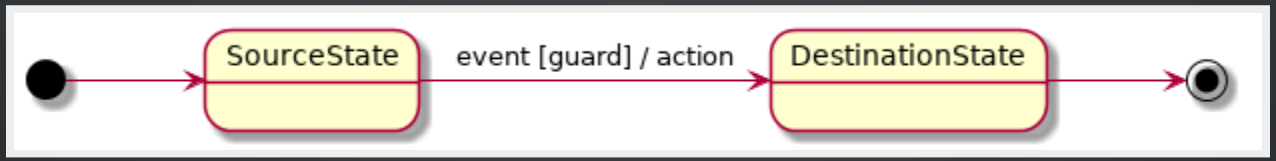



UNIFIED MODELING LANGUAGE (UML 2.5)

# CONNECTION - STATE DIAGRAM - TRANSITION




## TRANSITION

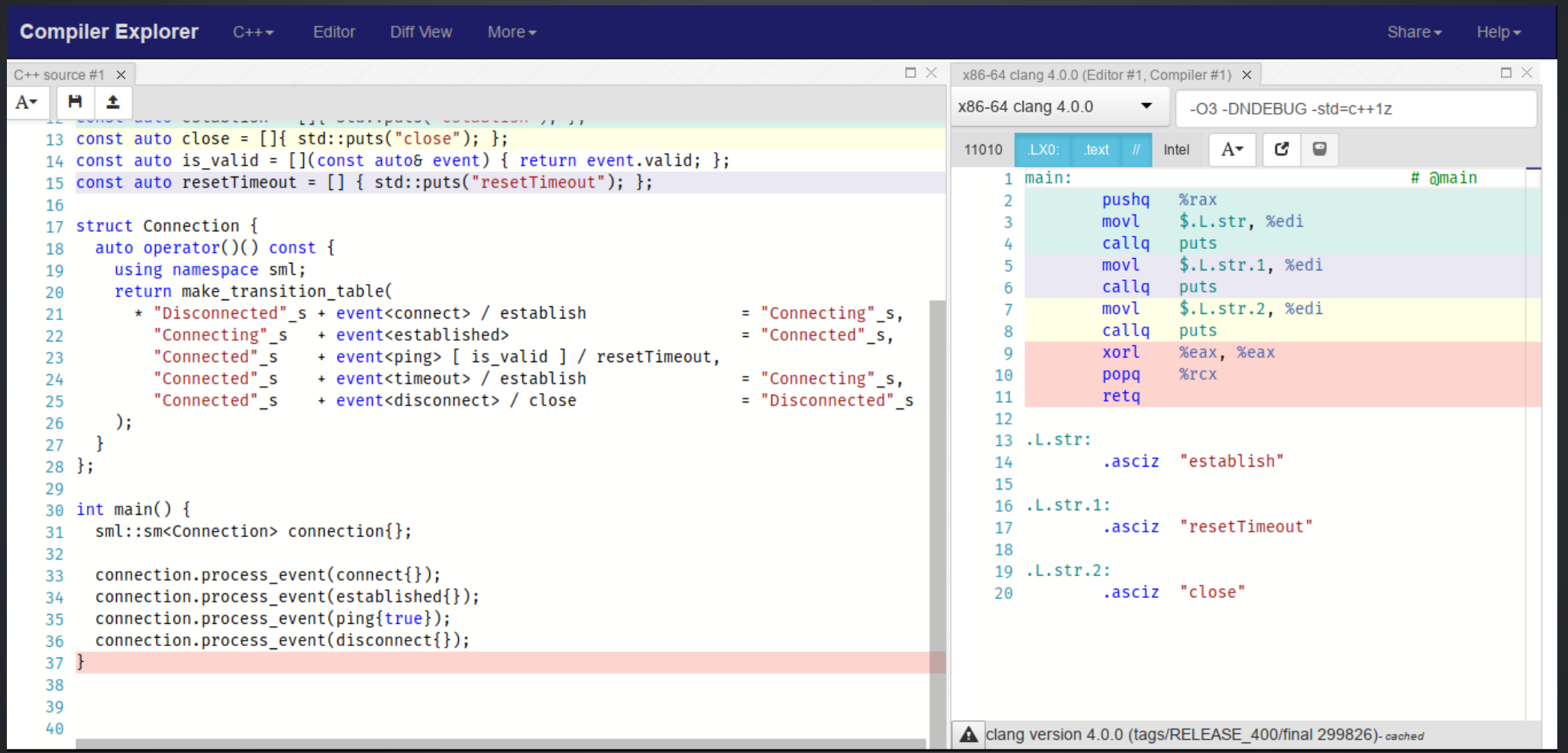


# [BOOST].SML - CONNECTION

```
// guards
const auto is_valid = [](auto event) { return event.valid; };

// actions
const auto establish = [] { std::puts("establish"); };
const auto close = [] { std::puts("close"); };
const auto resetTimeout = [] { std::puts("resetTimeout"); };

// events (types)
struct connect {}; // ...
struct ping { bool valid{}; };
```

```
int main() {
  sml::sm connection = [] {
    return transition_table{
      * "Disconnected"_s + connect / establish                = "Connecting"_s,
        "Connecting"_s   + established                        = "Connected"_s,
        "Connected"_s    + ping [ is_valid ] / resetTimeout,
        "Connected"_s    + timeout / establish                = "Connecting"_s,
        "Connected"_s    + disconnect / close                 = "Disconnected"_s
    };
  };

  sm.process_event(connect{});
}
```

# [BOOST].SML - PERFORMANCE



# BENCHMARK - RESULTS

| | Enum/Switch | Variant | [Boost].SML | Boost.MSM-eUML | Boost.Statechart |
|---|---|---|---|---|---|
| Compilation time | 0.132s | 15.321s | 0.582s | 1m15.935s | 5.671s |
| Execution time | 679ms | 827ms | 622ms | 664ms | 2282ms |
| Memory usage | 1b | 2b/8b | 1b | 120b | 224b |
| Executable size | 15K | 187K | 34K | 611K | 211K |
| Line of Code (LOC) | ~300 (no macros) | ~300 | ~50 | ~100 | ~200 |

# [BOOST].SML / QUESTIONS?

- One header - 2k LOC - (boost/sml.hpp) / generated
- Neither Boost nor STL is required
- Quick compilation-times (-Wall -Wextra -Werror -pedantic -pedantic-errors)
- Blazing fast run-time (Generated at compile-time)
- No 'virtual's (-fno-rtti)
- Optional support for 'exception's (-fno-exceptions)
- Supported compilers (C++14)
  - Clang-3.4+, XCode-6.1+, GCC-5.2+, MSVC-2015+

---

https://github.com/boost-experimental/sml