Fixing C++ With Epochs

Vittorio Romeo

vittorioromeo.info

vittorio.romeo@outlook.com

vromeo5@bloomberg.net

@supahvee1234

CppCon 2019

2019/09/17

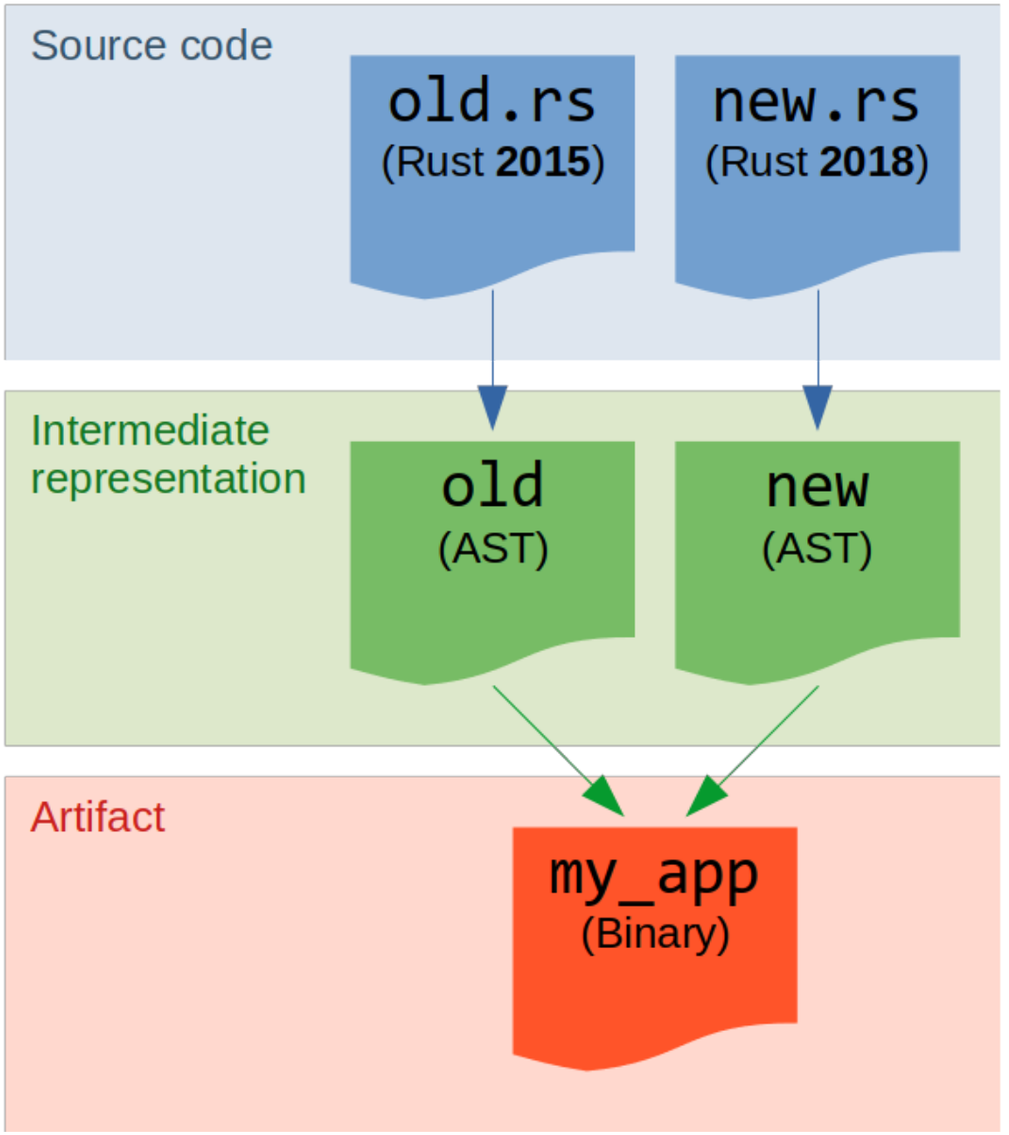
Aurora, CO

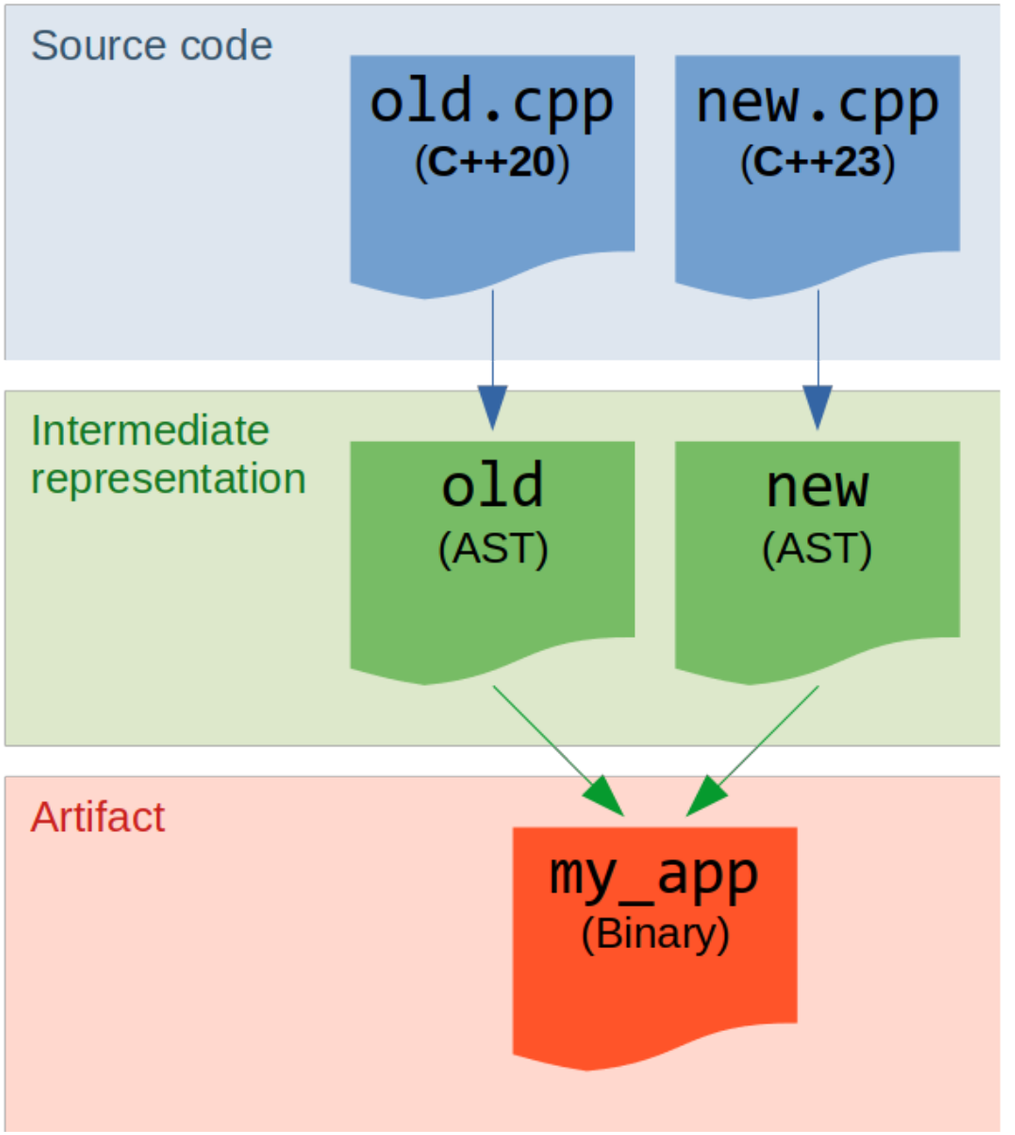
Bloomberg

- C++ has a problem, and it's only getting worse
  - Too lenient *(e.g. implicit conversions)*
  - Obsolete features *(e.g.* `typedef`*)*
  - Dangerous default *(e.g.* `[[nodiscard]]`, `explicit`*)*
  - Excessive complexity *(e.g. variable initialization)*
  - Unpleasantness *(e.g.* `co_await`*)*
  - ...

- Everybody wants to make C++ safer and simpler...
- ...but **backwards compatibility** is a key feature of the language

- Can we **remove** or **change** language features while preserving backwards compatibility?





- *Example:* let's get rid of implicit conversions in C++23
- We want the following to **fail compilation**

```
void serialize(stream&, int);

stream out;
serialize(out, 15.32f); // ⟸ should not compile!
```

- If desired, the user can always do:

```
serialize(out, static_cast<int>(15.32f));
```


- If we make the conversion ill-formed in C++23, we break code
- However, we can make this change opt-in via modules

```
module serialization;
using C++20;

void serialize(stream&, int);

void example()
{
    stream out;
    serialize(out, 15.3f); // OK
}
```

```
module serialization;
using C++23;

void serialize(stream&, int);

void example()
{
    stream out;
    serialize(out, 15.3f); // ERROR
}
```


- What I am proposing
  - Mechanism to clean up **language** syntax and features
  - **Linear** and **incremental** progression
  - Opt-in **backwards-compatible** system, module-level
  - **Simple** and **automatable** migration path
- What I am **not** proposing
  - ABI breakage
  - Many small tunable knobs *(creates dialects)*
  - More choices for users *(no, we want less choices!)*

- Benefits
  - C++ becomes appealing to newcomers
  - C++ becomes easier to teach
  - C++ becomes leaner
  - C++ becomes safer
  - C++ becomes more readable
  - C++ still looks like C++
  - Avoids community fragmentation
  - People can migrate easily, at their own pace

Thanks!

https://vittorioromeo.info

https://github.com/SuperV1234/cppcon2019

vittorio.romeo@outlook.com

vromeo5@bloomberg.net

@supahvee1234

Bloomberg

