



# Polymorphism Decision Table

---

Kilian Henneberger  
kilis-mail@web.de



# Last time at C++ London...

The screenshot shows a YouTube search results page for the query "no paradigm programming". The page layout includes a top navigation bar with the YouTube logo, a search bar containing the query, and utility icons for video, grid, and sign-in. On the left, there is a sidebar with navigation options: Home, Trending, Subscriptions, Library, and History. The main content area displays a list of search results, with the top result highlighted by a red border. The highlighted result is a video titled "No-Paradigm Programming" by Klaus Iglberger, with 385 views and streamed 2 weeks ago. The video thumbnail shows a streamer's interface with a QR code and a "Thank you!" message. Below the highlighted result are two other results: "[MUC++] Klaus Iglberger - Embrace No Paradigm Programming!" and "Lecture Collection | Programming Paradigms" by Stanford. The third result is "Bjarne Stroustrup - Object Oriented Programming without Inheritance - ECOOP 2015".

**YouTube** DE

no paradigm programming

**FILTER**

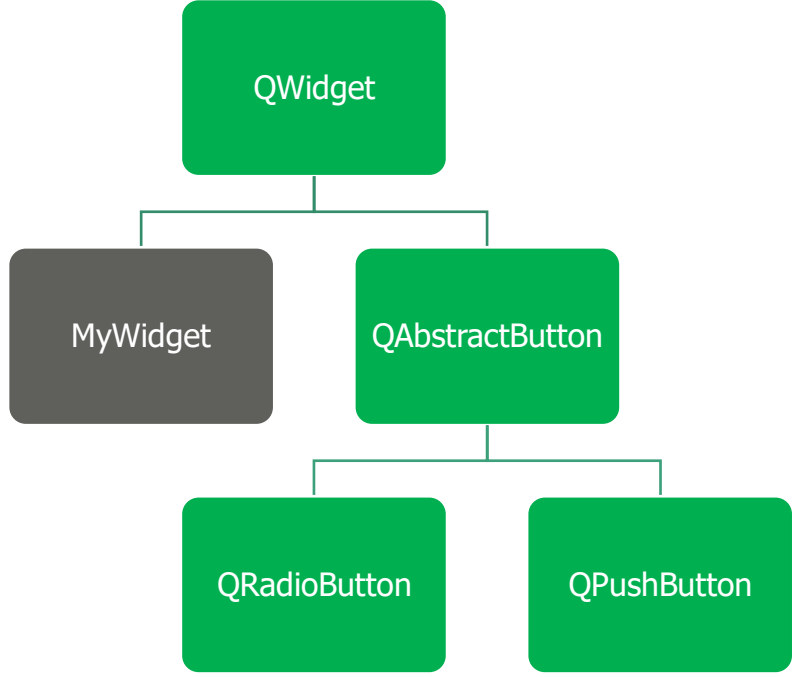
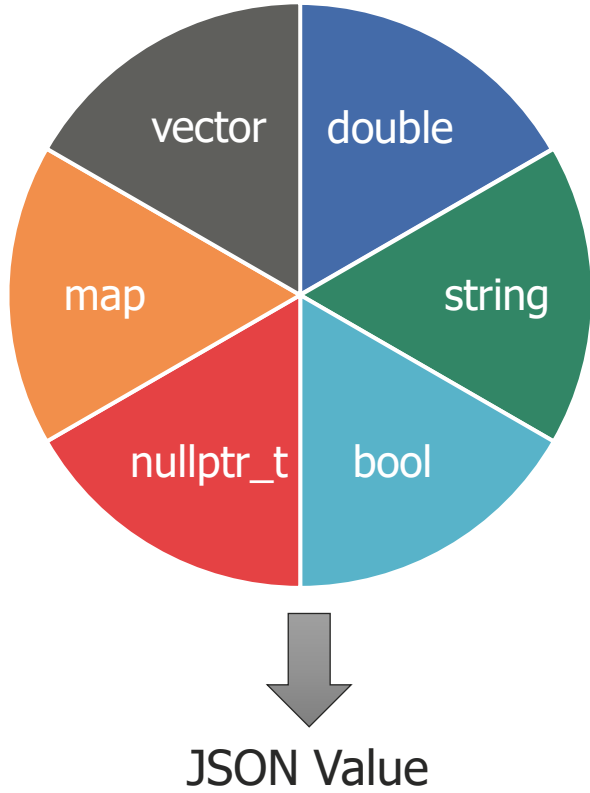
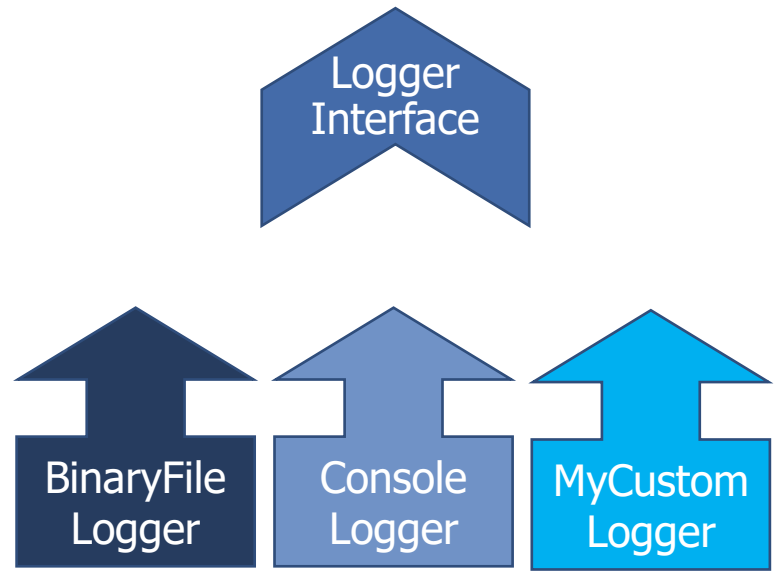
**No-Paradigm Programming**  
C++ London • 385 views • Streamed 2 weeks ago  
Main talk starts at: [https://youtu.be/t\\_0JhHVh\\_Kg?t=496](https://youtu.be/t_0JhHVh_Kg?t=496) C++ London moves online! "No-Paradigm Programming" - Klaus Iglberger ...

**[MUC++] Klaus Iglberger - Embrace No Paradigm Programming!**  
MUCplusplus • 529 views • 2 months ago  
What kind of language is C++? Is it a procedural programming language? An object-oriented programming language? A functional ...

**Lecture Collection | Programming Paradigms**  
Stanford  
Lecture 1 | Programming Paradigms (Stanford) • 17:26  
Lecture 2 | Programming Paradigms (Stanford) • 51:04  
VIEW FULL PLAYLIST

**Bjarne Stroustrup - Object Oriented Programming without Inheritance - ECOOP 2015**  
ECOOP! • 31K views • 4 years ago  
ECOOP, Prague July 9th 2015. <http://2015.ecoop.org> <http://curry-on.org>.

# Poly Morphs of Polymorphism



# Polymorphism Decision Table

Set of Types	CL	CL	CL	CL	CL	CL	CL	CL	OP	OP	OP	OP
Set of Functions	CL	CL	CL	CL	OP	OP	OP	OP	CL	CL	CL	CL
Semantics	VALUE	VALUE	REF	REF	VALUE	VALUE	REF	REF	VALUE	VALUE	REF	REF
Common Base-Class	NO	YES	NO	YES	NO	YES	NO	YES	NO	YES	NO	YES
Solution	<b>variant</b>	<b>variant, PV</b>	<b>TER, variant &lt;SP&gt;</b>	<b>SP</b>	<b>variant</b>	<b>PV+HVP, variant</b>	<b>variant &lt;SP&gt;</b>	<b>SP+HVP, variant &lt;SP&gt;</b>	<b>TEV</b>	<b>PV</b>	<b>TER</b>	<b>SP</b>

- CL closed
- OP open
- TEV type erasure with value semantics
- TER type erasure with reference semantics
- SP smart-pointer
- PV polymorphic\_value
- HVP hand-written visitor pattern

# Further Sources

- Klaus Iglberger “Embrace No-Paradigm Programming” C++ London  
[https://www.youtube.com/watch?v=t\\_0JhHVh\\_Kg](https://www.youtube.com/watch?v=t_0JhHVh_Kg)
- Bartek`s coding blog, “Everything You Need to Know About std::variant from C++17”  
<https://www.bfilipek.com/2018/06/variant.html>
- Sean Parent “Better Code: Runtime Polymorphism” NDC Conferences 2017  
<https://www.youtube.com/watch?v=QGcVXgEVMJg>
- Arthur O`Dwyer “Back To Basics: Type Erasure” CppCon 2019  
<https://www.youtube.com/watch?v=tbUCHifyT24>
- Jonathan Coe, polymorphic\_value  
[https://github.com/jbcoe/polymorphic\\_value](https://github.com/jbcoe/polymorphic_value)