

1 main — MIR Walkthrough

Purpose: TODO: Describe why this walkthrough exists

1.1 Source Context

```
fn main() {
    let a: WithParam<u32> = WithParam{the_t: 42, another: 42};

    let b: WithParam<u64> = WithParam{the_t: 42, another: 42};

    let c: Result<u8, usize> = Err(a.another);
    let d : Result<u64, u8> = Ok(b.the_t);

    let x = c.err().unwrap();

    assert!(x as u64 == d.unwrap());
}
```

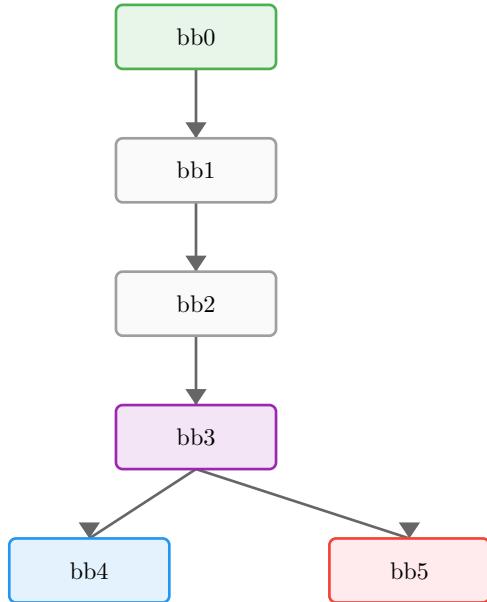
1.2 Function Overview

- **Function:** main
- **Basic blocks:** 6
- **Return type:** ()
- **Notable properties:**
 - Contains panic path
 - Has conditional branches

1.3 Locals

Local	Type	Notes
0	()	Return place
1	WithParam<u32>	
2	WithParam<u64>	
3	std::result::Result<u8, usize>	
4	usize	
5	std::result::Result<u64, u8>	
6	u64	
7	usize	
8	std::option::Option<usize>	
9	bool	
10	u64	
11	u64	
12	!	

1.4 Control-Flow Overview



1.5 Basic Blocks

1.5.1 bb0 — entry

Entry point of the function.

MIR	Annotation
<code>_1 = WithParam(42, 42)</code>	Construct aggregate
<code>_2 = WithParam(42, 42)</code>	Construct aggregate
<code>_4 = _1.1</code>	Copy value
<code>_3 = Result::Err(move _4)</code>	Construct aggregate
<code>_6 = _2.0</code>	Copy value
<code>_5 = Result::Ok(move _6)</code>	Construct aggregate
<code>→ _8 = err(_3) → bb1</code>	Call err

1.5.2 bb1

MIR	Annotation
<code>→ _7 = unwrap(move _8) → bb2</code>	Call unwrap

1.5.3 bb2

MIR	Annotation
<code>_10 = _7 as RigidTy(Uint(U64))</code>	Integer conversion
<code>→ _11 = unwrap(_5) → bb3</code>	Call unwrap

1.5.4 bb3 — branch point

MIR	Annotation
<code>_9 = move _10 == move _11</code>	Equal operation
<code>→ switch(move _9) \[0→bb5; else→bb4\]</code>	Branch on move _9

1.5.5 bb4 — return / success

Normal return path.

MIR	Annotation
→ <code>return</code>	Return from function

1.5.6 bb5 — panic path

Panic/diverging path.

MIR	Annotation
→ <code>_12 = panic(\[16 bytes\])</code>	Call panic

1.6 Key Observations

TODO: Add bullet points summarizing what this MIR teaches

-
-

1.7 Takeaways

TODO: One or two sentences to generalize this example

