

CONCURRENT WITNESS2TEST: Test-Harnessing the Power of Concurrency

Levente Bajczi

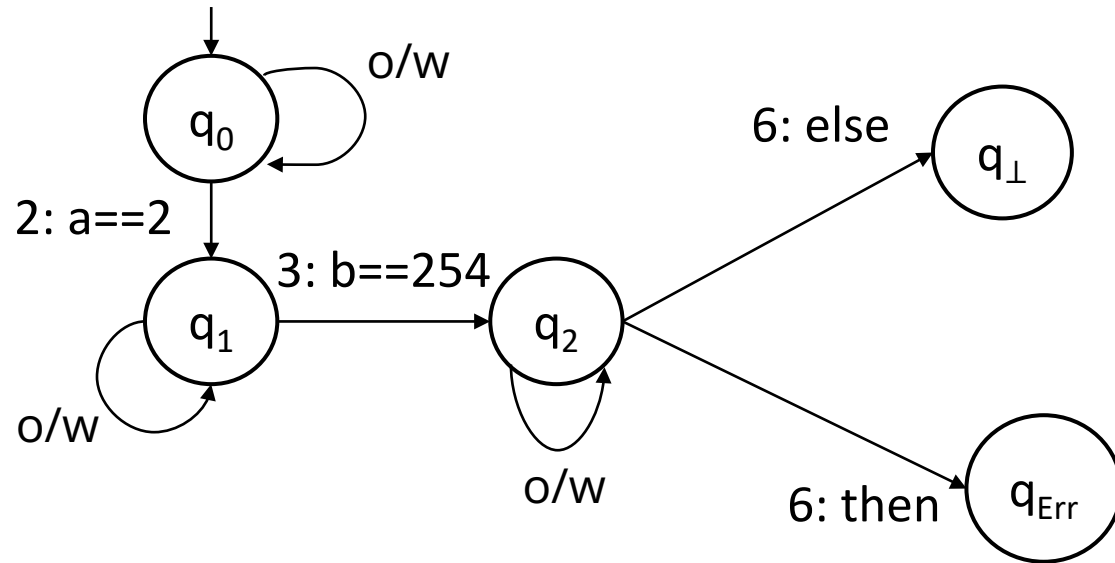
Zsófia Ádám

Zoltán Micskei



**Critical Systems
Research Group**

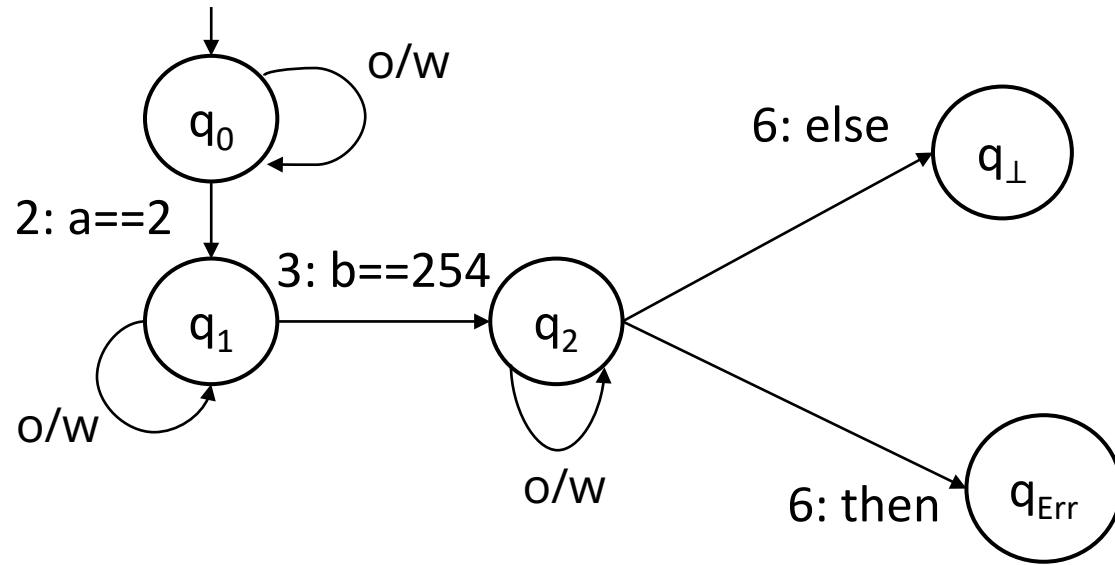
Test-harnesses in validation



(o/w = otherwise)

```
1  int main(void) {
2      unsigned char a = nondet_char();
3      unsigned char b = nondet_char();
4      unsigned char sum = a + b;
5      unsigned char mean = sum / 2;
6      if (mean < a / 2) {
7          error();
8      }
9      return 0;
10 }
```

Test-harnesses in validation

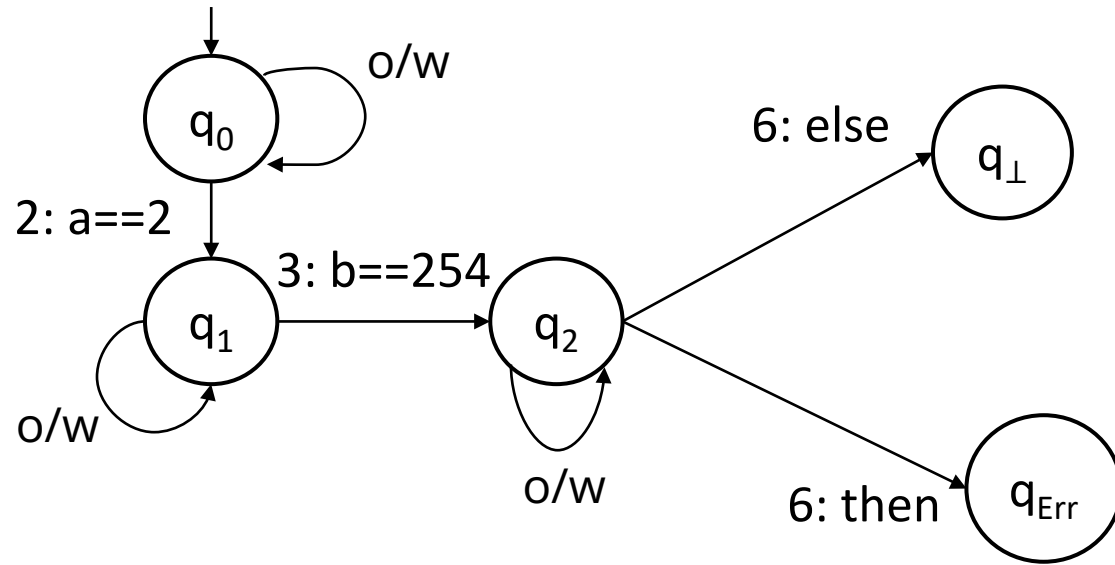


(o/w = otherwise)

```
1  int main(void) {
2      unsigned char a = nondet_char();
3      unsigned char b = nondet_char();
4      unsigned char sum = a + b;
5      unsigned char mean = sum / 2;
6      if (mean < a / 2) {
7          error();
8      }
9      return 0;
10 }
```

```
0  int idx = 0;
1  unsigned char nondet_char(void) {
2      switch(idx) {
3          case 0: ++idx; return 2;
4          case 1: ++idx; return 254;
5          default: return -1;
6      }
7  }
```


Test-harnesses in validation



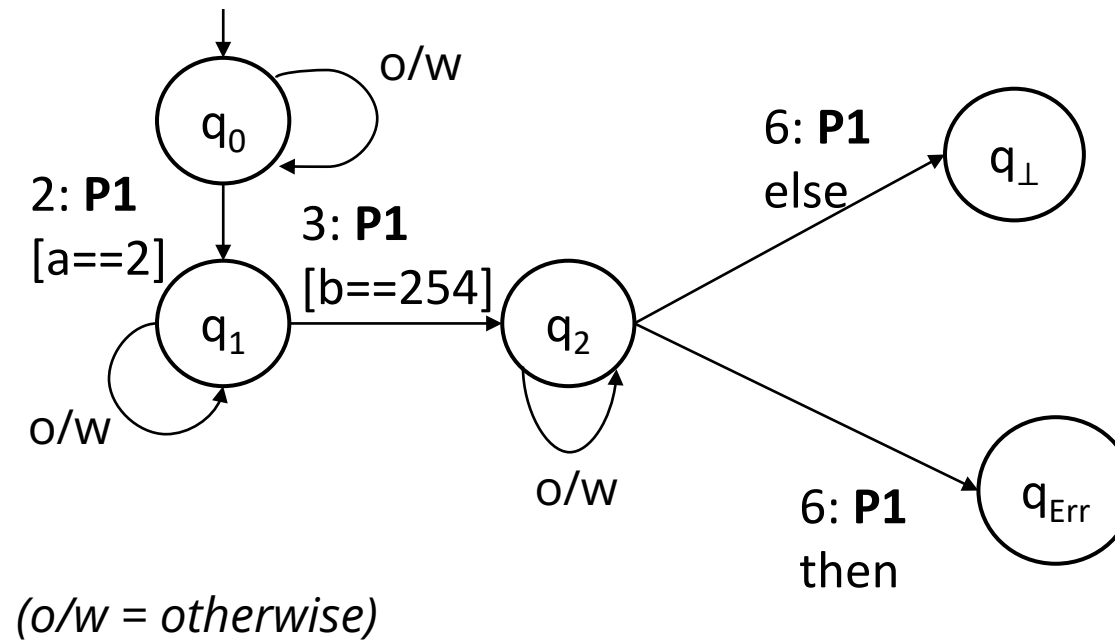
(o/w = otherwise)

**Executable
counterexample**

```
1 int main(void) {
2     unsigned char a = nondet_char();
3     unsigned char b = nondet_char();
4     unsigned char sum = a + b;
5     unsigned char mean = sum / 2;
6     if (mean < a / 2) {
7         error();
8     }
9     return 0;
10 }
```

```
0 int idx = 0;
1 unsigned char nondet_char(void) {
2     switch(idx) {
3         case 0: ++idx; return 2;
4         case 1: ++idx; return 254;
5         default: return -1;
6     }
7 }
```

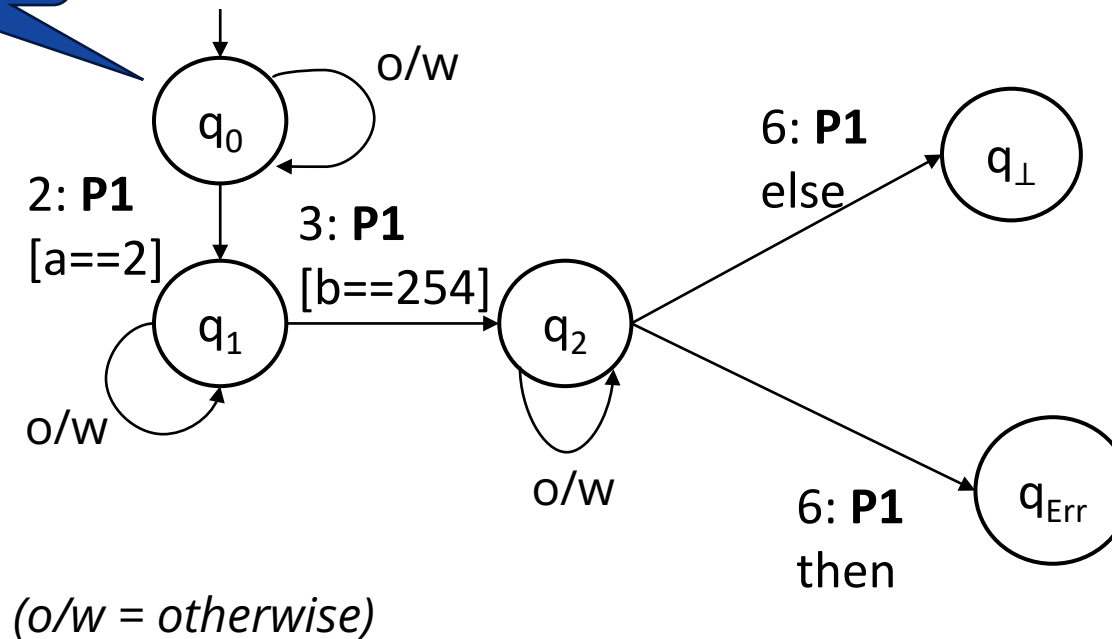
Multi-threaded programs and witnesses



Multi-threaded programs and witnesses

isEntry
isSink
isViolation

Nodes w/ info

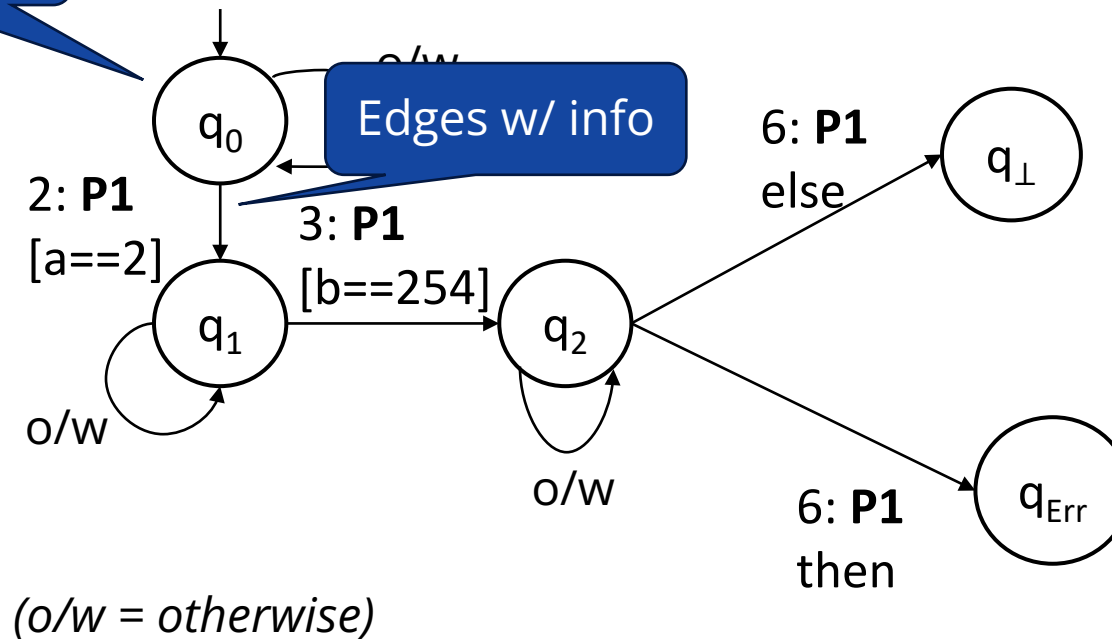


Multi-threaded programs and witnesses

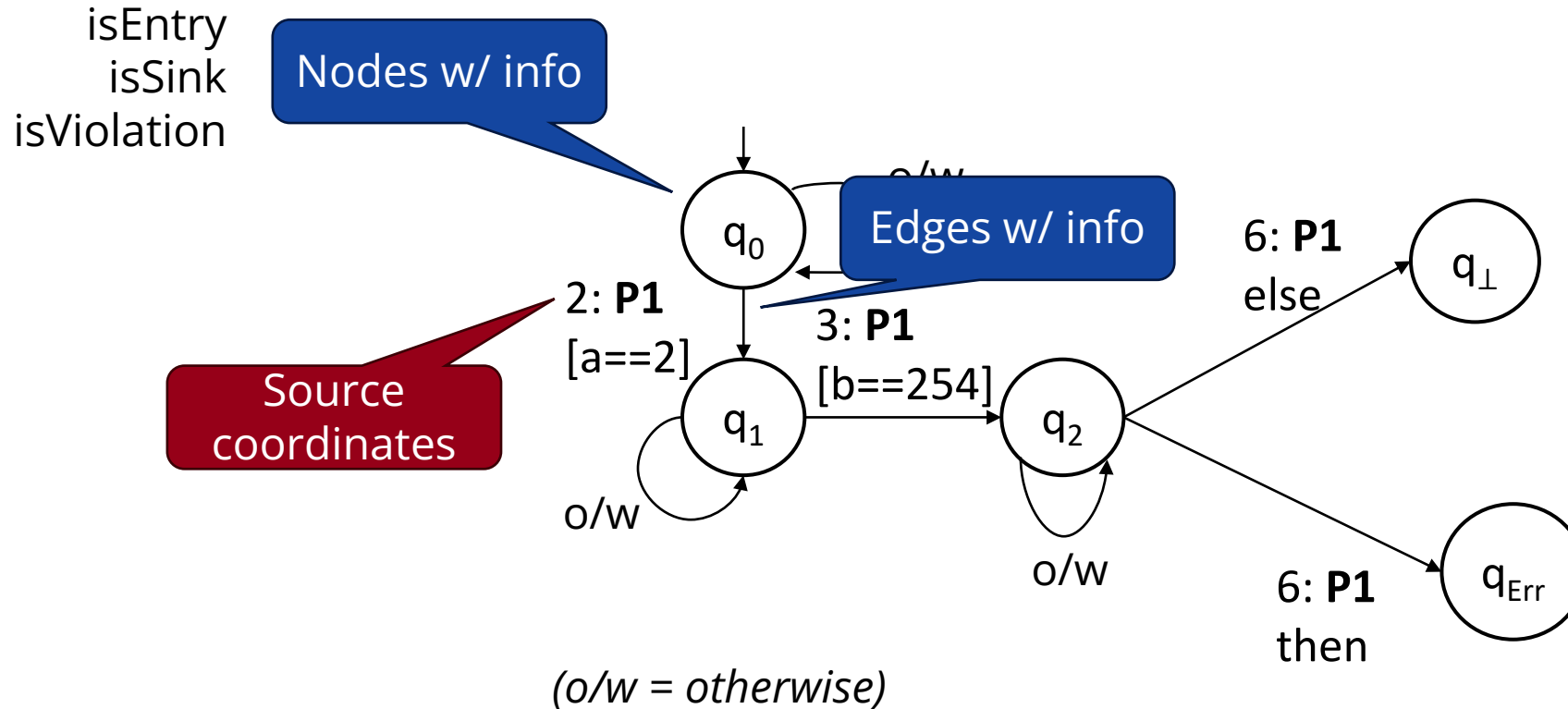
isEntry
isSink
isViolation

Nodes w/ info

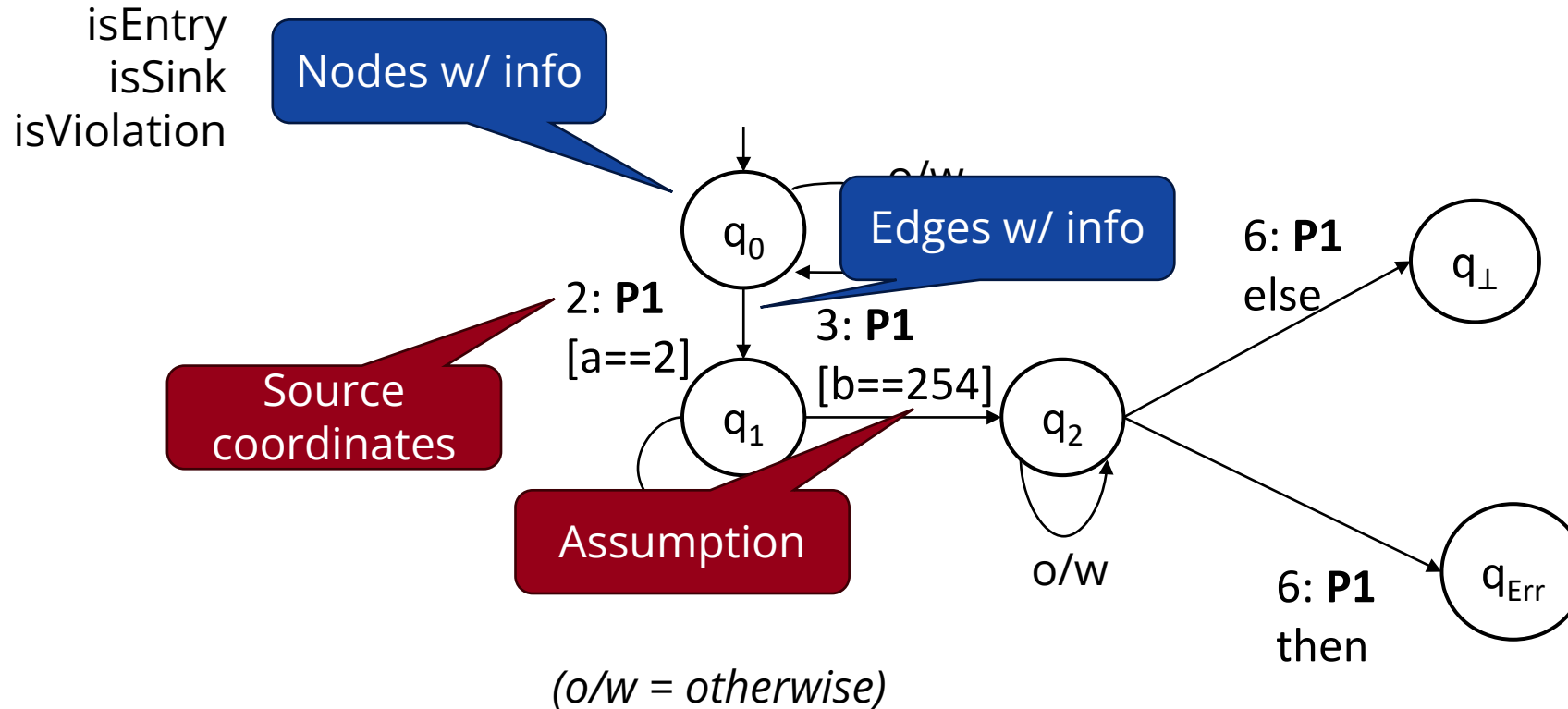
Edges w/ info



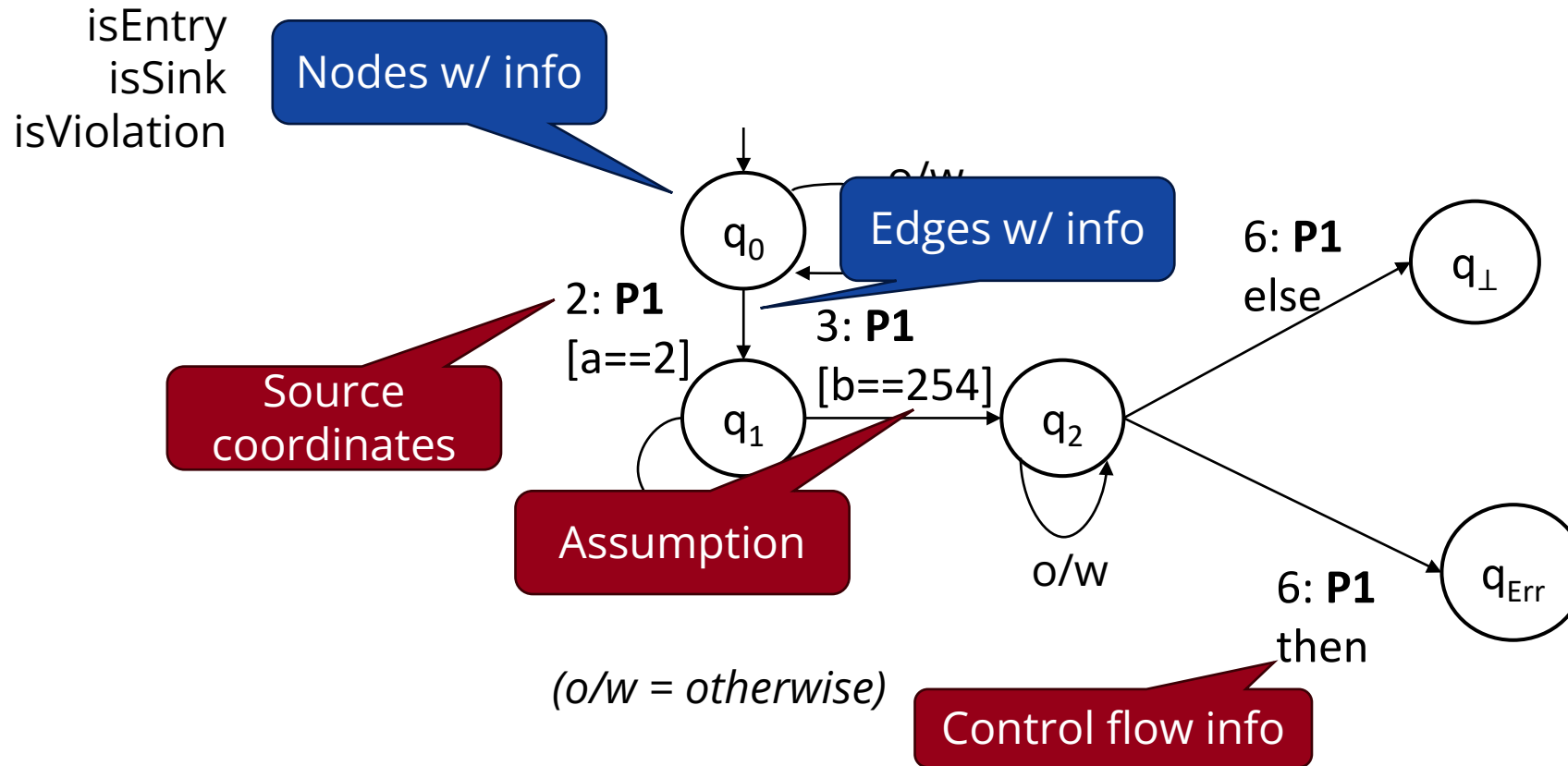
Multi-threaded programs and witnesses



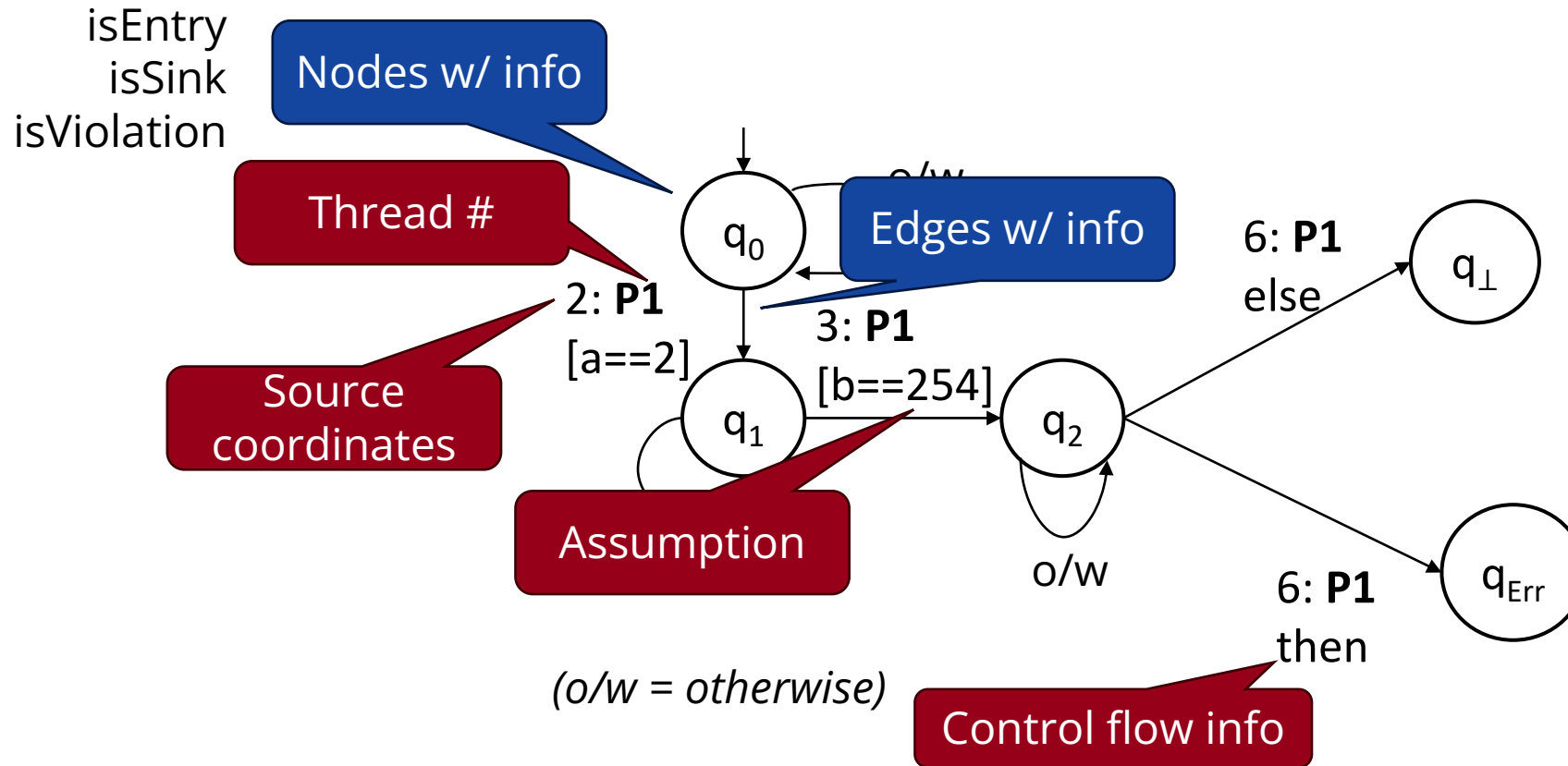
Multi-threaded programs and witnesses



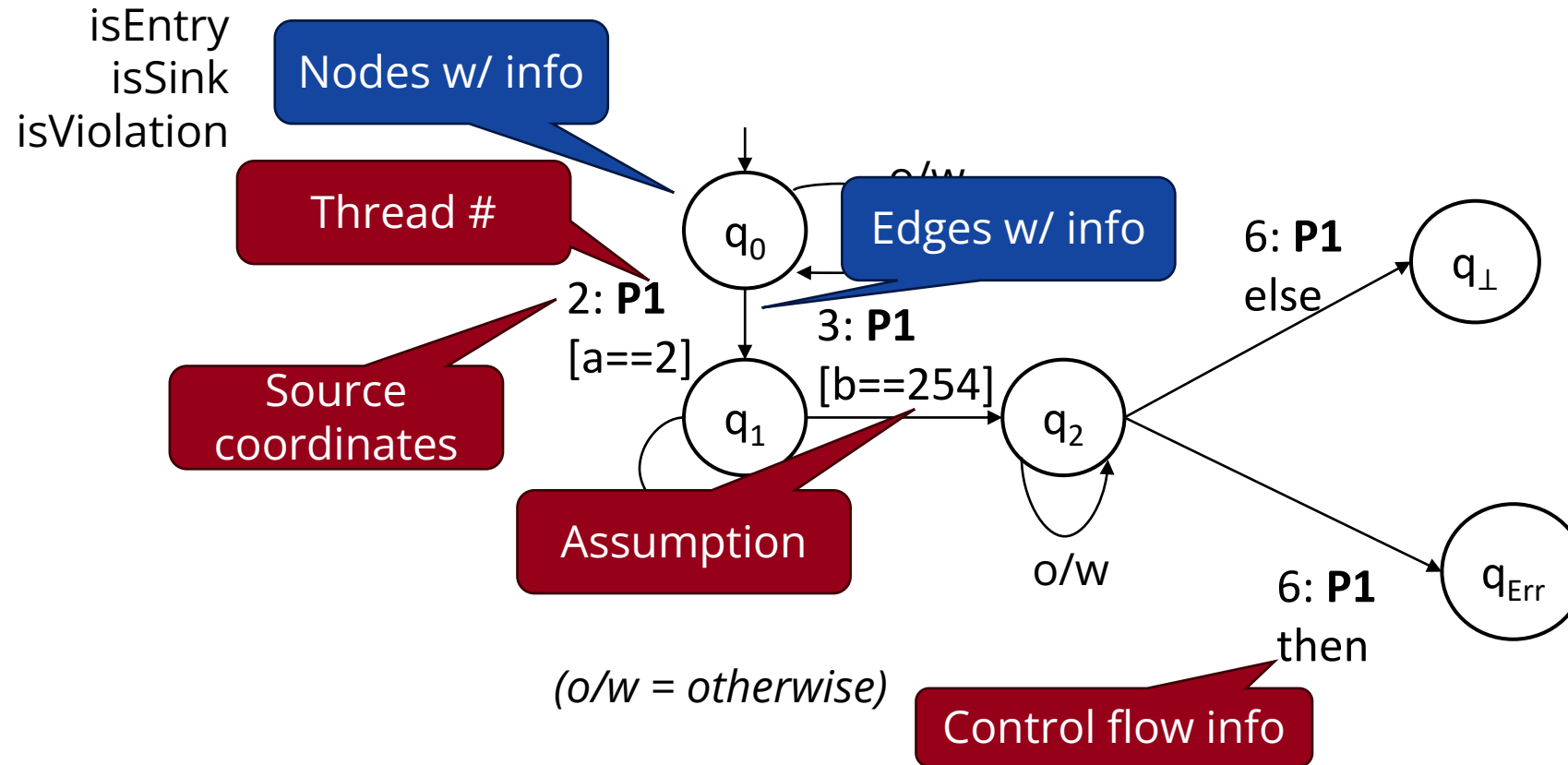
Multi-threaded programs and witnesses



Multi-threaded programs and witnesses



Multi-threaded programs and witnesses



Nothing is mandatory → **flexibility** for verifiers, **complexity** for validators

Handling concurrency

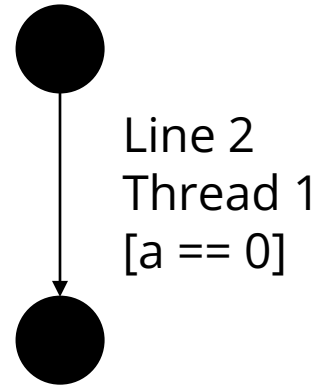
```
0  int a;

1  void P1(void) {
2      a = 0;
3      assert(a == 0);
4  }

5  void P2(void) {
6      a = 1;
7  }
```

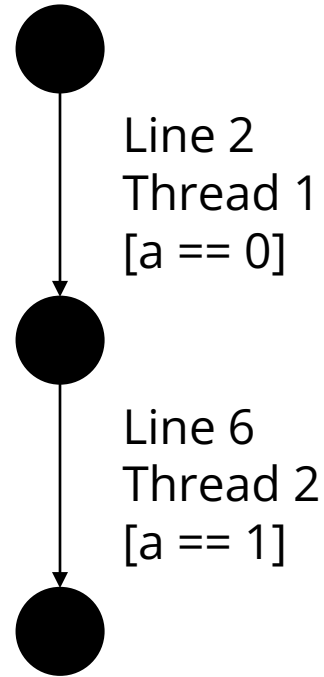
Handling concurrency

```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



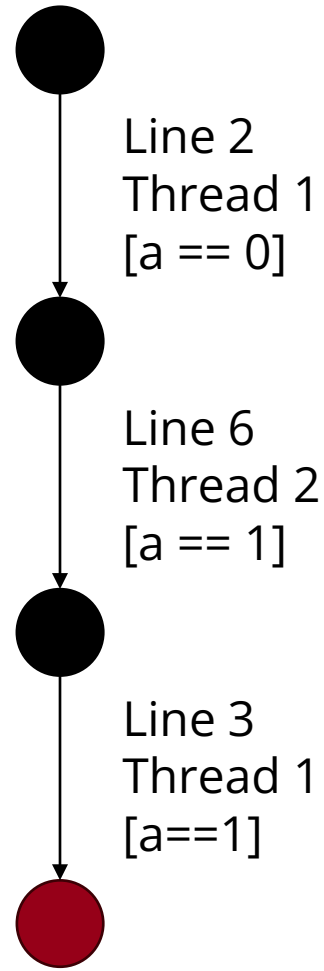
Handling concurrency

```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



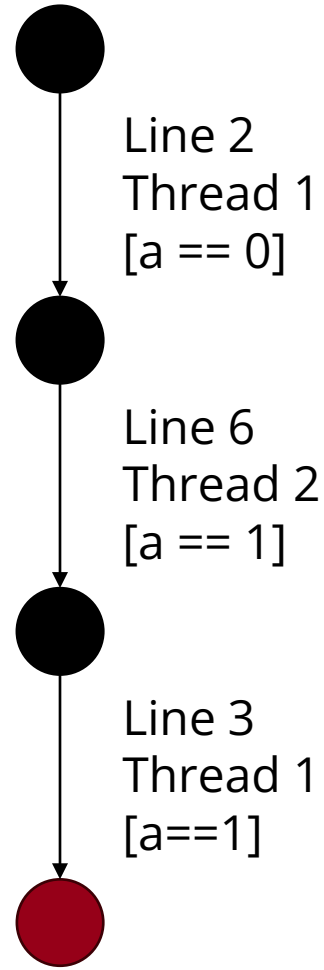
Handling concurrency

```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



Handling concurrency

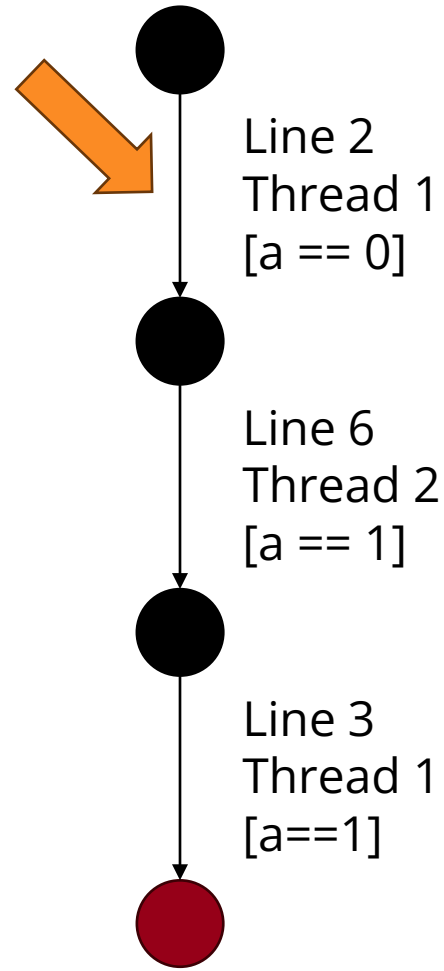
```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```

Handling concurrency

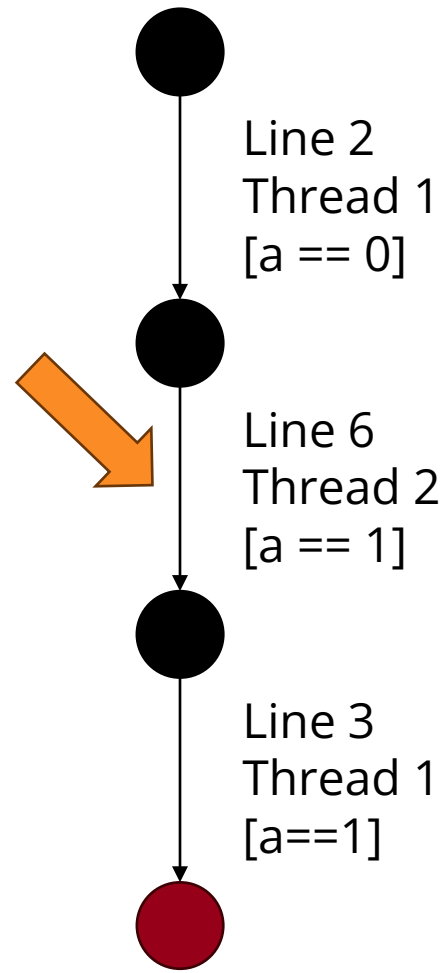
```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



```
0  int a;  
  
1  void P1(void) {  
2      yield(0);  
3      a = 0;  
4      release(0);  
5  }  
  
6  void P2(void) {  
7      a = 1;  
8  }
```

Handling concurrency

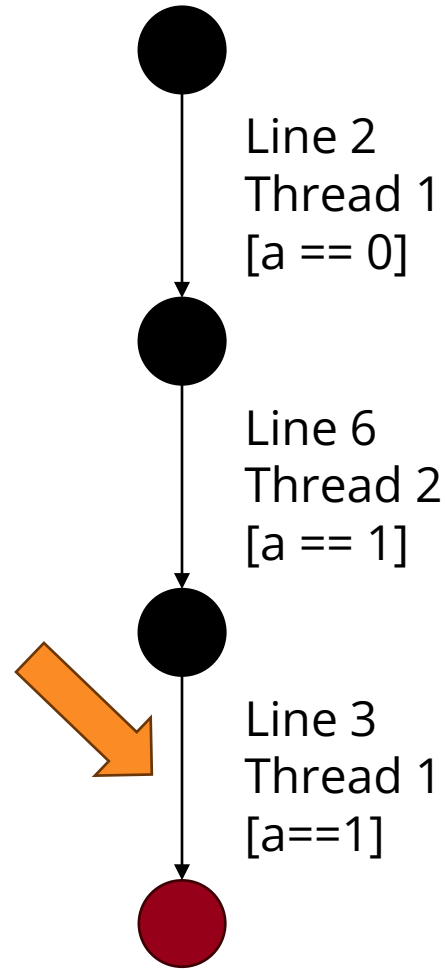
```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



```
0  int a;  
  
1  void P1(void) {  
2      yield(0);  
3      a = 0;  
4      release(0);  
5      assert(a == 0);  
6  }  
  
7  void P2(void) {  
8      yield(1);  
9      a = 1;  
10     release(1);  
11 }  
  
12 }
```

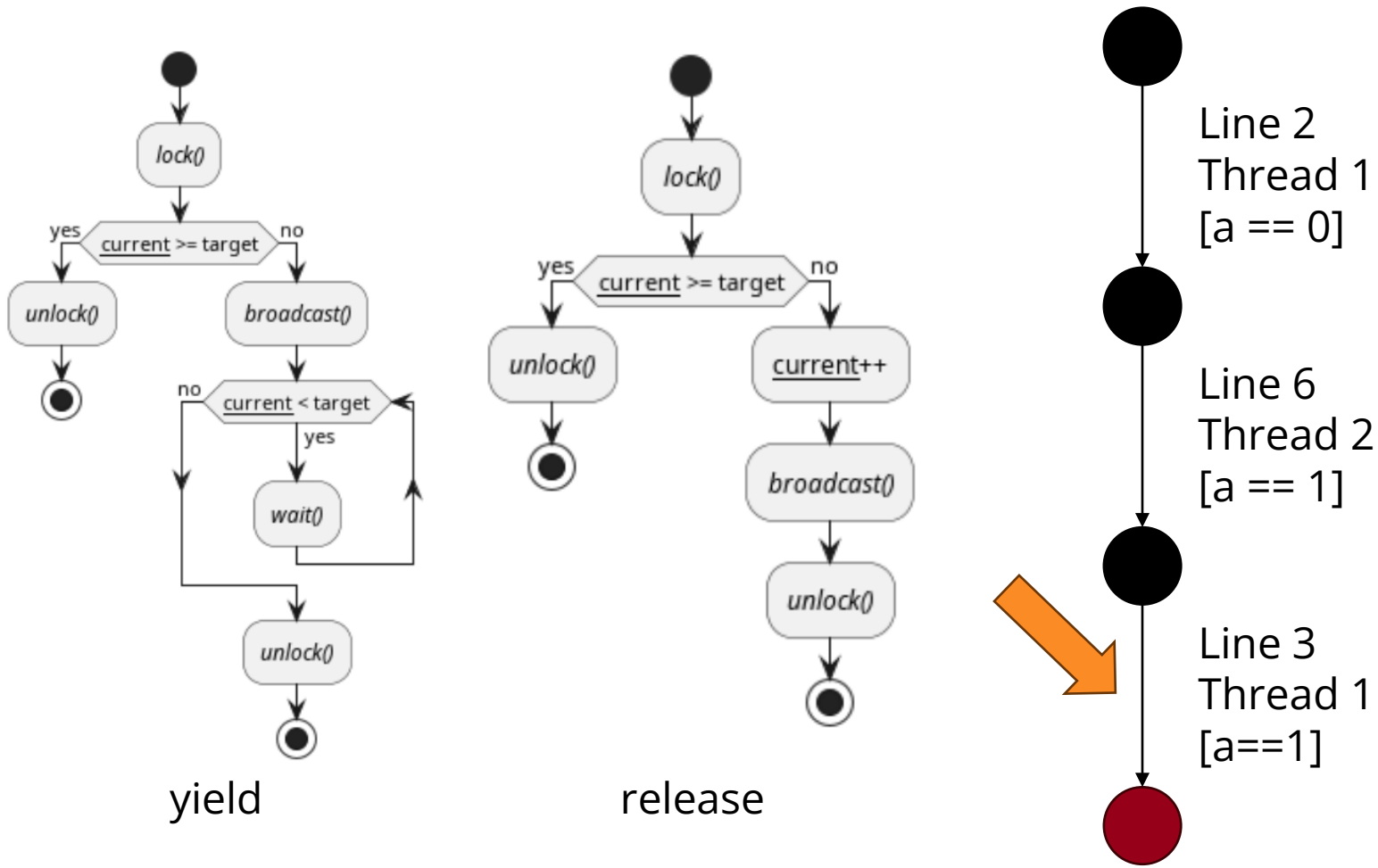
Handling concurrency

```
0  int a;  
  
1  void P1(void) {  
2      a = 0;  
3      assert(a == 0);  
4  }  
  
5  void P2(void) {  
6      a = 1;  
7  }
```



```
0  int a;  
  
1  void P1(void) {  
2      yield(0);  
3      a = 0;  
4      release(0);  
5      yield(2);  
6      assert(a == 0);  
7      release(2);  
8  }  
  
9  void P2(void) {  
10     yield(1);  
11     a = 1;  
12     release(1);  
13 }
```


Handling concurrency



```

0  int a;

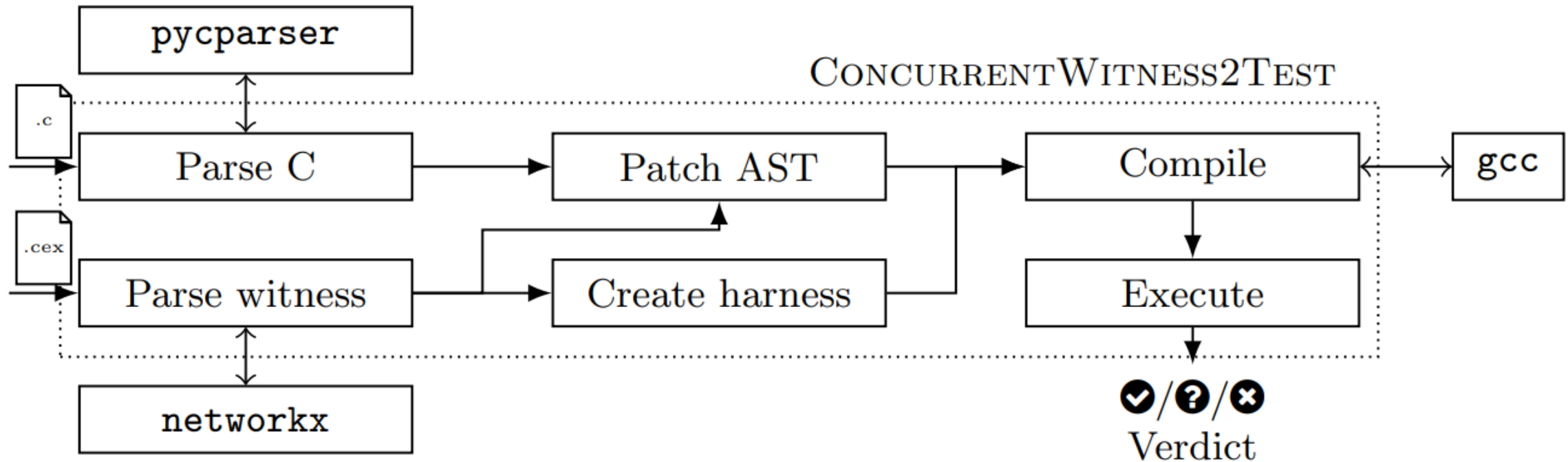
1  void P1(void) {
2      yield(0);
3      a = 0;
4      release(0);
5      yield(2);
6      assert(a == 0);
7      release(2);
8  }

9  void P2(void) {
10     yield(1);
11     a = 1;
12     release(1);
13 }

```

Software architecture

ftsrg/ConcurrentWitness2Test



Results

ftsrg/**ConcurrentWitness2Test**



	dartagnan	divine	theta	uautomizer	ugemcutter	utaipan
Confirmed	178	179 (2)	191	186	235	228
Refused	79	25 (1)	8	74	22	29
Error	193	111	96	168	194	170

Competition version



[10.5281/zenodo.10184336](https://doi.org/10.5281/zenodo.10184336)

Results

ftsrg/ConcurrentWitness2Test



	dartagnan	divine	theta	uautomizer	ugemcutter	utaipan
Confirmed	178	179 (2)	191	186	235	228
Refused	79	25 (1)	8	74	22	29
Error	193	111	96	168	194	170

Competition version



[10.5281/zenodo.10184336](https://doi.org/10.5281/zenodo.10184336)

Tool Support

bubaak	Unsupported (0/0/1)	cbmc	Supported (17/261/285)	coveriteam verifier algo selection	Supported (1/7/8)
coveriteam verifier parallel portfolio	Supported (19/230/258)	cpa lockator	Supported (39/9/90)	cpachecker	Supported (9/244/257)
cseq	Supported (290/18/656)	dartagnan	Supported (137/121/285)	deagle	Supported (50/7/585)
divine	Supported (136/195/374)	ebf	Supported (13/260/284)	esbmc incr	Supported (19/44/70)
esbmc kind	Supported (13/249/267)	goblint	Unsupported (0/0/41)	graves par	Supported (8/21/33)
graves	Supported (13/243/261)	infer	Supported (290/282/658)	lazycseq	Unknown (0/0/0)
if checker	Supported (7/12/291)	pesco	Supported (10/242/256)	pichecker	Supported (13/255/269)
symbiotic	Supported (16/87/112)	theta	Supported (2/34/37)	uautomizer	Supported (153/108/284)
ugemcutter	Supported (152/103/294)	utaipan	Supported (153/108/284)		

Now:

2 unsupported tools
23 supported tools

Results

ftsrg/ConcurrentWitness2Test



	dartagnan	divine	theta	uautomizer	ugemcutter	utaipan
Confirmed	178	179 (2)	191	186	235	228
Refused	79	25 (1)	8	74	22	29
Error	193	111	96	168	194	170

Competition version

Tool Support

bubaak	Unsupported (0/0/1)	cbmc	Supported (17/261/285)	coveriteam verifier algo selection	Supported (1/7/8)
coveriteam verifier parallel portfolio	Supported (19/230/258)	cpa lockator	Supported (39/9/90)	cpachecker	Supported (9/244/257)
cseq	Supported (290/18/656)	dartagnan	Supported (137/121/285)	deagle	Supported (50/7/585)
divine	Supported (136/195/374)	ebf	Supported (13/260/284)	esbmc incr	Supported (19/44/70)
esbmc kind	Supported (13/249/267)	goblint	Unsupported (0/0/41)	graves par	Supported (8/21/33)
graves	Supported (13/243/261)	infer	Supported (290/282/658)	lazycseq	Unknown (0/0/0)
if checker	Supported (7/12/291)	pesco	Supported (10/242/256)	pichecker	Supported (13/255/269)
symbiotic	Supported (16/87/112)	theta	Supported (2/34/37)	uautomizer	Supported (153/108/284)
ugemcutter	Supported (152/103/294)	utaipan	Supported (153/108/284)		

Now:

2 unsupported tools
23 supported tools

Most tools give underspecified witnesses:
not all paths lead to an error!



[10.5281/zenodo.10184336](https://doi.org/10.5281/zenodo.10184336)