

CRASH REPRODUCERS IN MLIR

Artemiy Bulavin

artemiyb@graphcore.ai



GRAPHCORE

WHAT THIS TALK IS ABOUT

Compilation pipelines can be complex and nuanced.

Crashes that result from buggy transformations become hard to track down.

Reproducing these bugs becomes hard.

What does MLIR have to help with this?

REPRODUCING BUGS 101

C++ unit test

```
Value of: isEven(16)
 Actual: false
Expected: true

[ FAILED ] MyIsEvenFunc.TestItWorks
[-----] 1 test from MyIsEvenFunc

[ FAILED ] 1 test, listed below:
[ FAILED ] MyIsEvenFunc.TestItWorks

1 FAILED TEST
```

```
TEST(MyIsEvenFunc, TestItWorks) {
    ASSERT_TRUE(isEven(16));
}
```

```
$ ctest -R TestIsEven
```

1. An output that you get from the (buggy) program

2. The output you expect to see

3. A way to reliably show that:

what you're getting != what you're expecting



REPRODUCING BUGS 101

Lit test

```
ctor<4xf32>, %arg1: vector<4xf32>, %arg2: f32) -> f32
xf32>
X error: no match found
  with "F" equal to "0"
  with "C" equal to "arg2"
: vector<4xf32> into f32
~~~~~
possible intended match
```

```
// CHECK-LABEL: func @extract_co
// CHECK-SAME: %[[A:.*0]]: vecto
// CHECK-SAME: %[[B:.*1]]: vecto
// CHECK-SAME: %[[C:.*2]]: f32
// CHECK:      %[[F:.*]] = arith
// CHECK:      %[[R:.*]] = vecto
// CHECK:      return %[[R]] : f
```

```
// RUN: mlir-opt %s \
//           --transform-interpreter \
//           | FileCheck %s
```

1. An output that you get from the (buggy) program

2. The output you expect to see

3. A way to reliably show that:
what you're getting != what you're expecting



REPRODUCING BUGS 101

Full compilation pipeline



User 16:17

My program won't compile :(



User 16:17

Everything works <3



Artemiy Bulavin 17:52

Please send me your code

what version are you using?

1. An output that you get from the (buggy) program

2. The output you expect to see

3. A way to ~~reliably~~ show that:

what you're getting != what you're expecting



EXAMPLE

Suppose we have a bug in the ConvertVectorToLLVM pass that won't lower `vector.contract` correctly:

```
mlir-opt --convert-vector-to-llvm  
vector-contract-to-dot-transforms.mlir
```

```
error: 'llvm.insertvalue' op Type mismatch: cannot insert 'f32' into '!llvm.array<2 x vector<2xf32>>'  
%res = vector.contract #matmat_trait %lhs, %rhs, %init  
^  
within split at vector-contract-to-dot-transforms.mlir:1 offset :201:10: note: see current operation:  
%39 = "llvm.insertvalue"(%3, %38) <{position = array<i64: 0>}> : (!llvm.array<2 x vector<2xf32>>,  
f32) -> !llvm.array<2 x vector<2xf32>>
```

EXAMPLE

Suppose we have a bug in the ConvertVectorToLLVM pass that won't lower `vector.contract` correctly:

We can tell `mlir-opt` to create a crash reproducer

```
mlir-opt --convert-vector-to-llvm  
--mlir-pass-pipeline-crash-reproducer=repro.mlir  
vector-contract-to-dot-transforms.mlir
```

```
error: 'llvm.insertvalue' op Type mismatch: cannot insert 'f32' into '!llvm.array<2 x vector<2xf32>>'  
%res = vector.contract #matmat_trait %lhs, %rhs, %init  
^  
within split at vector-contract-to-dot-transforms.mlir:1 offset :201:10: note: see current operation:  
%39 = "llvm.insertvalue"(%3, %38) <{position = array<i64: 0>}> : (!llvm.array<2 x vector<2xf32>>,  
f32) -> !llvm.array<2 x vector<2xf32>>  
within split at vector-contract-to-dot-transforms.mlir:1 offset :0:0: error: Failures have been  
detected while processing an MLIR pass pipeline  
within split at vector-contract-to-dot-transforms.mlir:1 offset :0:0: note: Pipeline failed while  
executing [`ConvertVectorToLLVMPass` on 'builtin.module' operation]: reproducer generated at  
`repro.mlir`
```

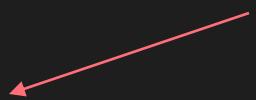
repro.mlir U X



repro.mlir U X

mlir > test > Dialect > Vector >  repro.mlir

The IR to reproduce



repro.mlir

```
mlir > test > Dialect > Vector > repro.mlir
1  #map = affine_map<(d0, d1, d2) -> (d0, d2)>
2  #map1 = affine_map<(d0, d1, d2) -> (d2, d1)>
3  #map2 = affine_map<(d0, d1, d2) -> (d0, d1)>
4  module {
5    func.func @contract_to_dot_matmat(%arg0: vector<2x2xf32>,
6                                     %arg1: vector<2x2xf32>,
7                                     %arg2: vector<2x2xf32>) -> vector<2x2xf32> {
8      %0 = vector.contract {
9        indexing_maps = [#map, #map1, #map2],
10       iterator_types = ["parallel", "parallel", "reduction"],
11       kind = #vector.kind<add>} %arg0, %arg1, %arg2 : vector<2x2xf32>, vector<2x2xf32> into vector<2x2xf32>
12      return %0 : vector<2x2xf32>
13    }
14  }
15
16 {-#
17   external_resources: {
18     mlir_reproducer: {
19       pipeline: "builtin.module(convert-vector-to-llvm{enable-amx=false enable-arm-i8mm=false enable-arm-neon=false enable-arm-sve=false enable-arm-vfpv4=false enable-memref=false enable-vec4=false enable-vec8=false enable-vec16=false enable-vec32=false enable-vec64=false enable-vec128=false enable-vec256=false enable-vec512=false enable-vec1024=false enable-vec2048=false enable-vec4096=false enable-vec8192=false enable-vec16384=false enable-vec32768=false enable-vec65536=false enable-vec131072=false enable-vec262144=false enable-vec524288=false enable-vec1048576=false enable-vec2097152=false enable-vec4194304=false enable-vec8388608=false enable-vec16777216=false enable-vec33554432=false enable-vec67108864=false enable-vec134217728=false enable-vec268435456=false enable-vec536870912=false enable-vec1073741824=false enable-vec2147483648=false enable-vec4294967296=false enable-vec8589934592=false enable-vec17179869184=false enable-vec34359738368=false enable-vec68719476736=false enable-vec137438953472=false enable-vec274877906944=false enable-vec549755813888=false enable-vec1099511627776=false enable-vec2199023255552=false enable-vec4398046511104=false enable-vec8796093022208=false enable-vec17592186044416=false enable-vec35184372088832=false enable-vec70368744177664=false enable-vec140737488355328=false enable-vec281474976710656=false enable-vec562949953421312=false enable-vec1125899906842640=false enable-vec2251799813685280=false enable-vec4503599627370560=false enable-vec9007199254741120=false enable-vec18014398509482240=false enable-vec36028797018964480=false enable-vec72057594037928960=false enable-vec144115188075857920=false enable-vec288230376151715840=false enable-vec576460752303431680=false enable-vec1152921504606863360=false enable-vec2305843009213726720=false enable-vec4611686018427453440=false enable-vec9223372036854906880=false enable-vec18446744073709813760=false enable-vec36893488147419627520=false enable-vec73786976294839255040=false enable-vec147573952589678510080=false enable-vec295147905179357020160=false enable-vec590295810358714040320=false enable-vec1180591620717428080640=false enable-vec2361183241434856161280=false enable-vec4722366482869712322560=false enable-vec9444732965739424645120=false enable-vec18889465931478849290240=false enable-vec37778931862957698580480=false enable-vec75557863725915397160960=false enable-vec151115727458230794321920=false enable-vec302231454916461588643840=false enable-vec604462909832923177287680=false enable-vec1208925819665846354575360=false enable-vec2417851639331692709150720=false enable-vec4835703278663385418301440=false enable-vec9671406557326770836602880=false enable-vec19342813114653541673205760=false enable-vec38685626229307083346411520=false enable-vec77371252458614166692823040=false enable-vec15474250491722833338564640=false enable-vec30948500983445666677129280=false enable-vec61897001966891333354258560=false enable-vec123794003933782666708517120=false enable-vec247588007867565333417034240=false enable-vec495176015735130666834068480=false enable-vec990352031470261333668136960=false enable-vec1980704062940522667336273920=false enable-vec3961408125881045334672547840=false enable-vec7922816251762090669345095680=false enable-vec15845632535324181338690191360=false enable-vec31691265070648362677380382720=false enable-vec63382530141296725354760765440=false enable-vec126765060282593450709521530880=false enable-vec253530120565186901419043061760=false enable-vec507060241130373802838086123520=false enable-vec1014120482260747605676172247040=false enable-vec2028240964521495211352344494080=false enable-vec4056481929042990422704688988160=false enable-vec8112963858085980845409377976320=false enable-vec16225927716171961690818755952640=false enable-vec32451855432343923381637511905280=false enable-vec64903710864687846763275023810560=false enable-vec129807421729375693526550447621120=false enable-vec259614843458751387053100895242240=false enable-vec519229686917502774106201790484480=false enable-vec1038459373835055548212403580968960=false enable-vec2076918747670111096424807161937920=false enable-vec4153837495340222192849614323875840=false enable-vec8307674990680444385699228647751680=false enable-vec1661534998136088877139845729550320=false enable-vec3323069996272177754279681459100640=false enable-vec6646139992544355508559362918201280=false enable-vec13292279985088711017118725836402560=false enable-vec26584559970177422034237451672805120=false enable-vec53169119940354844068474903345610240=false enable-vec10633823980670968813694806687220480=false enable-vec21267647961341937627389613374440960=false enable-vec42535295922683875254779226748881920=false enable-vec85070591845367750509558453497763840=false enable-vec170141183690735501019116856995527680=false enable-vec340282367381471002038233713991055360=false enable-vec680564734762942004076467427982110720=false enable-vec1361129469525884008152934855964221440=false enable-vec2722258939051768016305869711928442880=false enable-vec5444517878103536032611739423856885760=false enable-vec1088903575620707206522347885771377520=false enable-vec217780715124141441304469577154275040=false enable-vec435561430248282882608939154308550080=false enable-vec87112286049656576521787830861710160=false enable-vec174224572099313153043576601723420320=false enable-vec348449144198626306087153203446840640=false enable-vec696898288397252612174306406893681280=false enable-vec1393796576794505224348532013787362560=false enable-vec2787593153589010448697064027574725120=false enable-vec5575186307178020897394128055149450240=false enable-vec11150372614356041794788256110298900480=false enable-vec22300745228712083589576512220597800960=false enable-vec44601490457424167179153024441195601920=false enable-vec89202980914848334358306048882386403840=false enable-vec178405961829696668716612097764772807680=false enable-vec356811923659393337433224195529545615360=false enable-vec713623847318786674866448391058591231280=false enable-vec142724769463757334973289678211782462560=false enable-vec285449538927514669946579356423564925120=false enable-vec570899077855029339893158712846729850240=false enable-vec1141798155700058679786317425693459700480=false enable-vec2283596311400117359572634851386919400960=false enable-vec4567192622800234719145269702773838801920=false enable-vec9134385245600469438290539405547677603840=false enable-vec18268770491200938876581078811095355207680=false enable-vec36537540982401877753162157622190710415360=false enable-vec7307508196480375550632431524438142082720=false enable-vec14615016392960751101268630548876284165440=false enable-vec29230032785921502202537261097752568330880=false enable-vec58460065571843004405074522195505136617760=false enable-vec11692003114368600881014904439101027323520=false enable-vec23384006228737201762029808878202054667040=false enable-vec46768012457474403524059617756404109334080=false enable-vec9353602491494880704811923551280821868160=false enable-vec1870720498298976140962384710256164372320=false enable-vec3741440996597952281924769420512328744640=false enable-vec7482881993195904563849538841024657493280=false enable-vec1496576398639180912769867768204915486560=false enable-vec2993152797278361825539735536409830973120=false enable-vec5986305594556723651079471072819661946240=false enable-vec1197261118911344730215894214563932388480=false enable-vec2394522237822689460431788428127864776960=false enable-vec4789044475645378920863576856255729553920=false enable-vec9578088951290757841727153712511459107840=false enable-vec19156177902581515683454307425022918215680=false enable-vec38312355805163031366908614850045836431360=false enable-vec76624711610326062733817229700091672862720=false enable-vec153249423220652125467634459400183345755440=false enable-vec306498846441304250935268918800366681510880=false enable-vec612997692882608501870537837600733363021760=false enable-vec1225995385765217003741075675201466726043520=false enable-vec245199077153043400748215135040293345207040=false enable-vec490398154306086800396430270080586685404080=false enable-vec980796308612173600792860540161173370808160=false enable-vec1961592617224347201585721080322346741616320=false enable-vec392318523444869440317144216064469348332640=false enable-vec784637046889738880634288432128938696653280=false enable-vec1569274093779477761268576864257877393306560=false enable-vec3138548187558955522537153728515755586733120=false enable-vec627709637511791104507430745703151117466240=false enable-vec125541927502358220901486149140630223532480=false enable-vec251083855004716441802972298281260447065960=false enable-vec502167710009432883605944596562520894131920=false enable-vec100433542001865616721188919312540178823840=false enable-vec200867084003731233442377838625080357647680=false enable-vec401734168007462466884755677250160715335360=false enable-vec803468336014924933769511354500321430670720=false enable-vec1606936672029849867539026709000642861341440=false enable-vec3213873344059699735078053418001285722682880=false enable-vec6427746688119399470156106836002571445365760=false enable-vec12855493376238798940312213672005142890735360=false enable-vec25710986752477597880624427344001028571470720=false enable-vec51421973504955195761248854688002057142941440=false enable-vec102843947009850391522497709376004114285882880=false enable-vec20568789401970078304495541875200822217776560=false enable-vec40137578803940156608985883750401644443553120=false enable-vec8027515760788031321797176750080328888706240=false enable-vec1605503152157606262359433350016065777412480=false enable-vec3211006304315212524718866700032131555229600=false enable-vec642201260863042504943773340006426311059200=false enable-vec1284402521726085009887546680012852622198400=false enable-vec2568805043452170019775093360025705244396800=false enable-vec5137610086904340039550186680051410488793600=false enable-vec10275220173808680079100373360010280297972800=false enable-vec20550440347617360038200686680020560595945600=false enable-vec40100880695234720076400373360041121191891200=false enable-vec80201761390469440152800746720082242383782400=false enable-vec16040352278093888030560148944016448476756800=false enable-vec32080704556187776061120297888032896953513600=false enable-vec64161409112375552030240595776065793874073600=false enable-vec12832281822475110406048119553601315874757600=false enable-vec25664563644950220812096239107202631749515200=false enable-vec51329127289875541624192478214405280388780800=false enable-vec102658254579751083248384956428801056157561600=false enable-vec20531650915950216649676991285760211135123200=false enable-vec40163301831900433299353982571520422267546400=false enable-vec80326603663800866598707965143040844435092800=false enable-vec16065320732760173119741593028608168874093600=false enable-vec32130641465520346239483986057216337741873200=false enable-vec64261282931040692478967972114432665483746400=false enable-vec12852256586208394495735954422866533187547200=false enable-vec25664513172416788991471908845732666375954400=false enable-vec51314856344833577978943817691465333551873600=false enable-vec102629712689667555957887635382930667103745600=false enable-vec205259425379335111915775270765861334067531200=false enable-vec40151885075867022383155054073182266813507200=false enable-vec80303770151734044766310108146364533627144800=false enable-vec160607540303468089532650216292729064354289600=false enable-vec32121508060693617906530043258545812878579200=false enable-vec64243016121387235813060086517091635758558400=false enable-vec128486032242754471626120173034183271517156800=false enable-vec256972064485508943252240346068366543570313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200=false enable-vec40145925517681430920384139371434524414144800=false enable-vec80301851035362861840768278742869048828289600=false enable-vec160593102070725723681536557485738097654289600=false enable-vec32118620414145144736307311497147619535153600=false enable-vec64237240828290289472614622994347619070558400=false enable-vec12846840165658057894529245989095239075153600=false enable-vec256936803313161157890584919781904781410313600=false enable-vec513144068971017886504480692136731587551873600=false enable-vec102614813794203577300960348427346311035353600=false enable-vec205229627588407154601920696854672622070707200
```



The passes to reproduce



repro.mlir U X

```
mlir > test > Dialect > Vector > repro.mlir
1  #map = affine_map<(d0, d1, d2) -> (d0, d2)>
2  #map1 = affine_map<(d0, d1, d2) -> (d2, d1)>
3  #map2 = affine_map<(d0, d1, d2) -> (d0, d1)>
4  module {
5    func.func @contract_to_dot_matmat(%arg0: vector<2x2xf32>,
6                                     %arg1: vector<2x2xf32>,
7                                     %arg2: vector<2x2xf32>) -> vector<2x2xf32> {
8      %0 = vector.contract {
9        indexing_maps = [#map, #map1, #map2],
10       iterator_types = ["parallel", "parallel", "reduction"],
11       kind = #vector.kind<add>} %arg0, %arg1, %arg2 : vector<2x2xf32>, vector<2x2xf32> into vector<2x2xf32>
12      return %0 : vector<2x2xf32>
13    }
14  }
15
16 {-#
17   external_resources: {
18     mlir_reproducer: {
19       pipeline: "builtin.module(convert-vector-to-llvm{enable-amx=false enable-arm-i8mm=false enable-arm-neon=false enable-arm-sve=false enable-arm-vfpv4=false enable-memref=false enable-vec4=false enable-vec8=false enable-vec16=false enable-vec32=false enable-vec64=false enable-vec128=false enable-vec256=false enable-vec512=false enable-vec1024=false enable-vec2048=false enable-vec4096=false enable-vec8192=false enable-vec16384=false enable-vec32768=false enable-vec65536=false enable-vec131072=false enable-vec262144=false enable-vec524288=false enable-vec1048576=false enable-vec2097152=false enable-vec4194304=false enable-vec8388608=false enable-vec16777216=false enable-vec33554432=false enable-vec67108864=false enable-vec134217728=false enable-vec268435456=false enable-vec536870912=false enable-vec1073741824=false enable-vec2147483648=false enable-vec4294967296=false enable-vec8589934592=false enable-vec17179869184=false enable-vec34359738368=false enable-vec68719476736=false enable-vec137438953472=false enable-vec274877906944=false enable-vec549755813888=false enable-vec1099511627776=false enable-vec2199023255552=false enable-vec4398046511104=false enable-vec8796093022208=false enable-vec17592186044416=false enable-vec35184372088832=false enable-vec70368744177664=false enable-vec140737488355328=false enable-vec281474976710656=false enable-vec562949953421312=false enable-vec1125899906842640=false enable-vec2251799813685280=false enable-vec4503599627370560=false enable-vec9007199254741120=false enable-vec18014398509482240=false enable-vec36028797018964480=false enable-vec72057594037928960=false enable-vec144115188075857920=false enable-vec288230376151715840=false enable-vec576460752303431680=false enable-vec1152921504606863360=false enable-vec2305843009213726720=false enable-vec4611686018427453440=false enable-vec9223372036854906880=false enable-vec18446744073709813760=false enable-vec36893488147419627520=false enable-vec73786976294839255040=false enable-vec147573952589678510080=false enable-vec295147905179357020160=false enable-vec590295810358714040320=false enable-vec1180591620717428080640=false enable-vec2361183241434856161280=false enable-vec4722366482869712322560=false enable-vec9444732965739424645120=false enable-vec18889465931478849290240=false enable-vec37778931862957698580480=false enable-vec75557863725915397160960=false enable-vec151115727458230794321920=false enable-vec302231454916461588643840=false enable-vec604462909832923177287680=false enable-vec1208925819665846354575360=false enable-vec2417851639331692709150720=false enable-vec4835703278663385418301440=false enable-vec9671406557326770836602880=false enable-vec19342813114653541673205760=false enable-vec38685626229307083346411520=false enable-vec77371252458614166692823040=false enable-vec15474250491722833338564640=false enable-vec30948500983445666677129280=false enable-vec61897001966891333354258560=false enable-vec123794003933782666708517120=false enable-vec247588007867565333417034240=false enable-vec495176015735130666834068480=false enable-vec990352031470261333668136960=false enable-vec1980704062940522667336273920=false enable-vec3961408125881045334672547840=false enable-vec7922816251762090669345095680=false enable-vec15845632535324181338690191360=false enable-vec31691265070648362677380382720=false enable-vec63382530141296725354760765440=false enable-vec126765060282593450709521530880=false enable-vec253530120565186901419043061760=false enable-vec507060241130373802838086123520=false enable-vec1014120482260747605676172247040=false enable-vec2028240964521495211352344494080=false enable-vec4056481929042990422704688988160=false enable-vec8112963858085980845409377976320=false enable-vec16225927716171961690818755952640=false enable-vec32451855432343923381637511905280=false enable-vec64903710864687846763275023810560=false enable-vec129807421729375693526550447621120=false enable-vec259614843458751387053100895242240=false enable-vec519229686917502774106201790484480=false enable-vec1038459373835055548212403580968960=false enable-vec2076918747670111096424807161937920=false enable-vec4153837495340222192849614323875840=false enable-vec8307674990680444385699228647751680=false enable-vec1661534998136088877139845729550320=false enable-vec3323069996272177754279681459100640=false enable-vec6646139992544355508559362918201280=false enable-vec13292279985088711017118725836402560=false enable-vec26584559970177422034237451672805120=false enable-vec53169119940354844068474903345610240=false enable-vec10633823980670968813694806687220480=false enable-vec21267647961341937627389613374440960=false enable-vec42535295922683875254779226748881920=false enable-vec85070591845367750509558453497763840=false enable-vec170141183690735501019116856995527680=false enable-vec340282367381471002038233713991055360=false enable-vec680564734762942004076467427982110720=false enable-vec1361129469525884008152934855964221440=false enable-vec2722258939051768016305869711928442880=false enable-vec5444517878103536032611739423856885760=false enable-vec1088903575620707206522347885771377520=false enable-vec217780715124141441304469577154275040=false enable-vec435561430248282882608939154308550080=false enable-vec87112286049656576521787830861710160=false enable-vec174224572099313153043576601723420320=false enable-vec348449144198626306087153203446840640=false enable-vec696898288397252612174306406893681280=false enable-vec1393796576794505224348532013787362560=false enable-vec2787593153589010448697064027574725120=false enable-vec5575186307178020897394128055149450240=false enable-vec11150372614356041794788256110298900480=false enable-vec22300745228712083589576512220597800960=false enable-vec44601490457424167179153024441195601920=false enable-vec89202980914848334358306048882387203840=false enable-vec178405961829696668716612097764774407680=false enable-vec356811923659393337433224195529548815360=false enable-vec713623847318786674866448391058597630720=false enable-vec142724769463757334973289678211795261440=false enable-vec285449538927514669946579356423590522880=false enable-vec570899077855029339893158712846781055760=false enable-vec1141798155700058679786317425693562111520=false enable-vec2283596311400117359572634851387124223040=false enable-vec4567192622800234719145269702774248446080=false enable-vec9134385245600469438290539405548496892160=false enable-vec18268770491200938876581078811096993784320=false enable-vec36537540982401877753162157622193987568640=false enable-vec73075081964803755506324315244387975137280=false enable-vec14615016392960751101268630488875955274560=false enable-vec29230032785921502202537260977751910549120=false enable-vec58460065571843004405074521955503830598240=false enable-vec116920031143686008810149043911007661196480=false enable-vec233840062287372017620298087822015332392960=false enable-vec467680124574744035240596175644034667858880=false enable-vec935360249149488070481192351288069335777760=false enable-vec1870720498298976140962384702576138671555520=false enable-vec374144099659795228192476940515227343511040=false enable-vec748288199319590456384953880230454667022080=false enable-vec1496576398639180912769867760460913340441760=false enable-vec2993152797278361825539735520891826670835520=false enable-vec5986305594556723651079471041783653341671040=false enable-vec1197261118911344730215894208356310668342080=false enable-vec2394522237822689460431788416672621336684160=false enable-vec4789044475645378920863576833345242673368320=false enable-vec9578088951290757841727153666690845346736640=false enable-vec19156177902581515683454307333381690693473280=false enable-vec38312355805163031366908614666763381386946560=false enable-vec76624711610326062733817229333526762773893120=false enable-vec15324942322065212546763445866705333554778240=false enable-vec30649884644130425093526891733402666709556480=false enable-vec61299769288260850187053783466805333511132960=false enable-vec12259953857652170037410756693361333302265920=false enable-vec24519907715304340074821513386722666704511840=false enable-vec49039815430608680079643026773445333308537680=false enable-vec98079630861217360079326053546890666717155360=false enable-vec1961592617224347200786521070937813333435111280=false enable-vec3923185234448694400773042141875626667070222560=false enable-vec7846370468897388800756084283751253333405445120=false enable-vec1569274093779477600753016856750250667010890240=false enable-vec3138548187558955200746033713500501333301780480=false enable-vec627709637511791040074306742700100266700360960=false enable-vec1255419275023582080073535585400200533300721920=false enable-vec25108385500471641600731711708004010667001443840=false enable-vec50216771000943283200728573416008021333002887680=false enable-vec10043354200186566400727146832016042667005775360=false enable-vec20086708400373132800726303664032085333001150720=false enable-vec40173416800746265600725607332064170667002301440=false enable-vec80346833600148531200725214664128340333004602880=false enable-vec16069366720029706240072512932856680667009205760=false enable-vec321387334400594124800725058657133613330018411520=false enable-vec642774668800118249600725029314267226670036823040=false enable-vec1285549337600236499200725014628534453330073646080=false enable-vec2571098675200472998400725007315768906670014730160=false enable-vec51421973504009459968007250036315378013330029460320=false enable-vec102843947008018919936007250018157756026670058920640=false enable-vec2056878940160378398720072500090355120533300117841280=false enable-vec4113757880320756797440072500045177560106700235682560=false enable-vec822751576064151359488007250002258878402133300471365120=false enable-vec1645503140128302789776007250001129435680426700942730240=false enable-vec32910062802566055795520072500005647178408533001885460480=false enable-vec65820125605132111591040072500002823588817067003770920960=false enable-vec131640251210264223822080072500001411794434133000754184160=false enable-vec26328050242052844764416007250000070589788682670003778368320=false enable-vec526561004841056895288320072500000352949577365330001886736640=false enable-vec10531220096821137905766400725000001764747886733300003773473280=false enable-vec210624401936422758115328007250000008823739443466700001886736640=false enable-vec4212488038728455162306560072500000044118697333333000003773473280=false enable-vec84249760774569103246131200725000000220593486667333000001886736640=false enable-vec16849952154913820649262400725000000110296743333330000003773473280=false enable-vec336999043098276412985248007250000000551483766673330000001886736640=false enable-vec6739980861965528259704960072500000002757418833333300000003773473280=false enable-vec134799617239310565194099200725000000013787094166733300000001886736640=false enable-vec2695992344786211303881984007250000000068935470833333000000003773473280=false enable-vec53919846895724226077639680072500000000344677354166733000000003773473280=false enable-vec107839693791448452155279360072500000000172338677333330000000003773473280=false enable-vec2156793875828969043105587200725000000000861693386667330000000003773473280=false enable-vec4313587751657938086211174400725000000000430846694133330000000001886736640=false enable-vec86271755033158761724223488007250000000002154233472667330000000003773473280=false enable-vec17254351006631752344844697600725000000000107711693633330000000001886736640=false enable-vec345087020132635046896893952007250000000000538558468166733000000003773473280=false enable-vec690174040265270093793787904007250000000000269279234083330000000001886736640=false enable-vec1380348080530540187587555880072500000000001346396170416673000000003773473280=false enable-vec2760696161061080375175111760072500000000000673198085208330000000001886736640=false enable-vec55213923221221607503502235200725000000000003365990426041673000000003773473280=false enable-vec110427846442432015067004467600725000000000001682995213020833000000001886736640=false enable-vec22085569288486403013400893520072500000000000084149760651041673000000003773473280=false enable-vec44171138576972806026801787040072500000000000042074880325520833000000001886736640=false enable-vec883422771539456012536035740800725000000000000210374401627604167300000003773473280=false enable-vec17668455430789102507207148160072500000000000010518720081380833000000001886736640=false enable-vec3533691086157820501441429632007250000000000000525936004069604167300000003773473280=false enable-vec706738217231564025288285926400725000000000000026296800203480833000000001886736640=false enable-vec1413476434463128050576571852800725000000000000013148400101960416730000003773473280=false enable-vec2826952868926256025153143705600725000000000000006574200050980833000000001886736640=false enable-vec5653905737852512050306287402400725000000000000003287100025490416730000003773473280=false enable-vec11307811475705024050612574704800725000000000000001643550012745083300000001886736640=false enable-vec226156229514100481012255494096007250000000000000008217750063730416730000003773473280=false enable-vec452312459028200962024510988192007250000000000000004108875031865083300000001886736640=false enable-vec904624918056401924048219776384007250000000000000002054437563730416730000003773473280=false enable-vec1809249836112803848096439552768007250000000000000001027218781865083300000001886736640=false enable-vec361849967222560769619287910553600725000000000000000051360939093304167300000003773473280=false enable-vec7236999344451215392385758527680072500000000000000002568046954665083300000001886736640=false enable-vec1447399868890243078477117785360072500000000000000001284023477330416730000003773473280=false enable-vec2894799737780486156954223585360072500000000000000000642011738665083300000001886736640=false enable-vec57895994755609723139084471707200725000000000000000003210058714330416730000003773473280=false enable-vec115791989511215446278168943414400725000000000000000001605029357330416730000001886736640=false enable-vec231583979022430892556337886828800725000000000000000000802514678665083300000003773473280=false enable-vec463167958044861785112755773657600725000000000000000000401257339330416730000001886736640=false enable-vec926335916089723570225511547315200725000000000000000000200628669666508330000003773473280=false enable-vec18526718321794471404477557736576007250000000000000000001003143348330416730000001886736640=false enable-vec37053436643588942808955115473152007250000000000000000000501571674166508330000003773473280=false enable-vec741068732871778856179102273630400725000000000000000000025078583708330416730000001886736640=false enable-vec148213746574355711235804454731520072500000000000000000001253929185166508330000003773473280=false enable-vec2964274931487114224716088594664007250000000000000000000062696459258330416730000001886736640=false enable-vec5928549862974228449432177189328007250000000000000000000031348229628330416730000003773473280=false enable-vec11857099725948456898863554378640072500000000000000000000156741148141667300000001886736640=false enable-vec237141994518969137977271087572800725000000000000000000000783705740728330416730000003773473280=false enable-vec47428398903793827595454217514560072500000000000000000000039185287036416730000001886736640=false enable-vec9485679780758765519090843502912007250000000000000000000001959264351828330416730000003773473280=false enable-vec1897135956151753103818168700582400725000000000000000000000097963217591416730000001886736640=false enable-vec37942719123035062076363374001164800725000000000000
```



- Reproducers can be created using `mlir-opt` and the PassManager
- Reproducers are **external resources**.
- Handled by AsmParser

```
{-# dialect_resources: {
      builtin: {
        blob1: "0x080000001000000000000000200000000000",
      },
      external_resources: {
        external: {
          blob: "0x080000001000000000000000200000000000",
        },
        other_stuff: {
          bool: true
        }
      }
    #-}
```

EXAMPLE

Once you have the reproducer, you can run it:

```
mlir-opt --run-reproducer repro.mlir
```

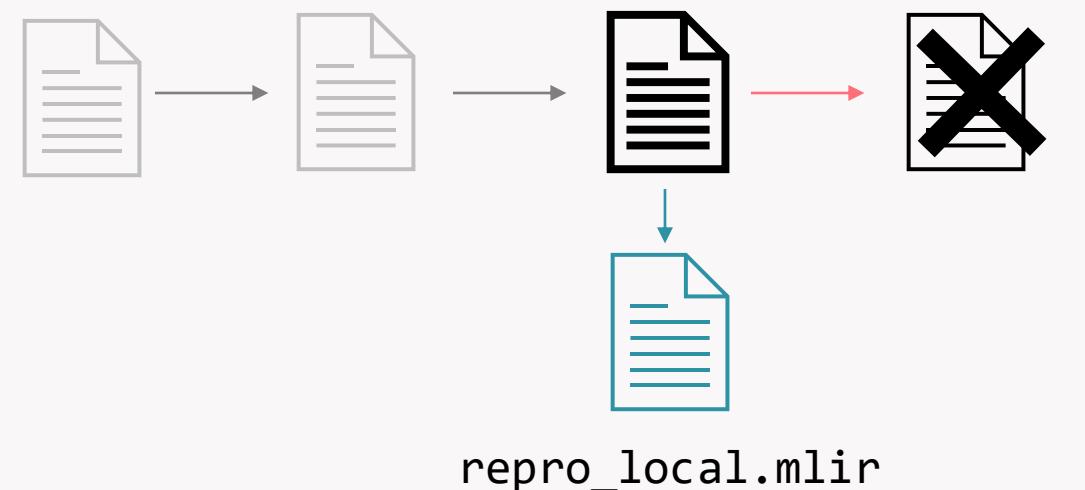
```
error: 'llvm.insertvalue' op Type mismatch: cannot insert 'f32' into '!llvm.array<2 x vector<2xf32>>'  
%res = vector.contract #matmat_trait %lhs, %rhs, %init  
      ^  
within split at vector-contract-to-dot-transforms.mlir:1 offset :201:10: note: see current operation:  
%39 = "llvm.insertvalue"(%3, %38) <{position = array<i64: 0>}> : (!llvm.array<2 x vector<2xf32>>,  
f32) -> !llvm.array<2 x vector<2xf32>>
```

LOCAL VS FULL REPRODUCERS

Reproducers contain the initial IR when you run the passes.



Local reproducers contain the IR and passes *just before failure*.



Our ConvertVectorToLLVM lowering contains a bug that affects `vector.contract`

What will **full** and **local** the reproducers for this IR look like?

(Demo)

```
#matmat_accesses = [
    // ...
]
#matmat_trait = {
    // ...
}
#map0 = // ...

func.func @contract_to_dot_matmat(
    %memref_lhs: memref<?x?xf32>, %rhs: vector<2x2xf32>,
    %init: vector<2x2xf32>) -> vector<2x2xf32> {
    %c0 = arith.constant 0 : index
    %cst = arith.constant 0.0 : f32

    %lhs = vector.transfer_read %memref_lhs[%c0, %c0], %cst { permutation_map=#map0 } :
        memref<?x?xf32>, vector<2x2xf32>

    %res = vector.contract #matmat_trait %lhs, %rhs, %init
        : vector<2x2xf32>, vector<2x2xf32> into vector<2x2xf32>
    return %res : vector<2x2xf32>
}
```



Use the PassManager:

- `PassManager::enableCrashReproducerGeneration(outputFile, generateLocal);`
- Analogous to `--mlir-pass-pipeline-crash-reproducer` and
`--mlir-pass-pipeline-local-reproducer`
- `mlir::makeReproducer(anchorName, passes, op, outputFile);`
- Analogous to `--mlir-generate-reproducer`

Let's look at an example out in the wild: **Triton**

EXAMPLES OUT IN THE WILD

```
auto reproducerPath =
    triton::tools::getStrEnv("TRITON_REPRODUCER_PATH");
if (!reproducerPath.empty()) {
    auto anchorName = self.getOpAnchorName();
    auto passes = self.getPasses();
    Operation *op = mod.getOperation();
    // Save a reproducer for the current pass manager invocation
    // immediately.
    makeReproducer(anchorName, passes, op, reproducerPath);
    // But if the pass manager crashes, attempt to generate a local
    // reproducer instead.
    context->disableMultithreading();
    self.enableCrashReproducerGeneration(reproducerPath,
        /*genLocalReproducer=*/true);
} else {
    self.enableCrashReproducerGeneration(makeConsoleReproducer());
}
// ... Later on, PassManager::run
```



Reduce the problem size:

- Remove passes that have no effect
- Remove ‘safe’ passes like canonicalisation, CSE
- Remove ops and values while still seeing the bug

If still unclear:

- Use a full reproducer, not local

USING REPRODUCERS WITH MLIR-REDUCE

- `mlir-reduce` is a tool for reducing the size of input IR that contains a bug.
- Iteratively applies transformations to the input to ‘reduce’ it, while checking the bug remains.
- You write an ‘interestingness script’, telling `mlir-reduce` whether the reduced IR contains the bug or not.
- `mlir-reduce` trims the input till it’s not ‘interesting’ anymore.

INTERESTINGNESS SCRIPT

Put the pipeline from the reproducer here

```
mlir-opt "$1" --pass-pipeline="..." 2>&1 | \  
    grep -q "cannot insert 'f32' into '!llvm.array"
```

```
# Exit code 1 if grep finds the error, 0 otherwise.  
if [ $? -eq 0 ]; then  
    exit 1  
else  
    exit 0  
fi
```

```
$ mlir-reduce --reduction-tree='traversal-mode=0 test=interesting.sh'  
reproducer.mlir
```

INTERESTINGNESS SCRIPT

```
mlir-opt "$1" --my-broken-pass-1 --my-broken-pass-2 ...

if [ $? -neq 0 ]; then
    exit 1
else
    exit 0
fi
```

```
$ mlir-reduce --reduction-tree='traversal-mode=0 test=interesting.sh'
reproducer.mlir
```

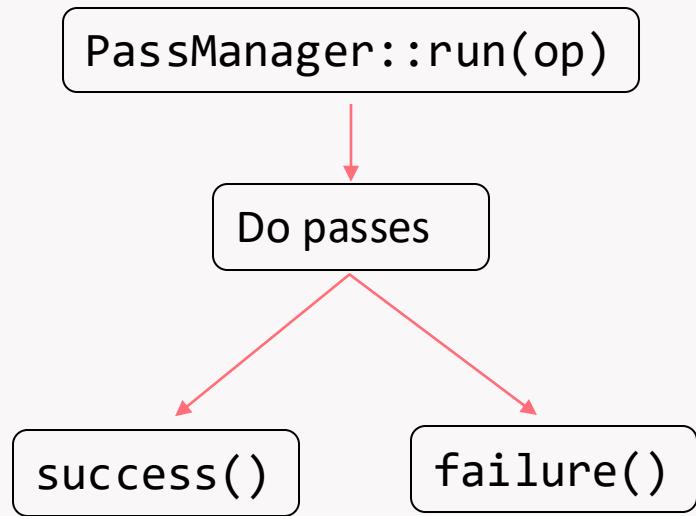
INTERESTINGNESS SCRIPT

```
mlir-runner "$1" ... 2>&1 | grep "error"

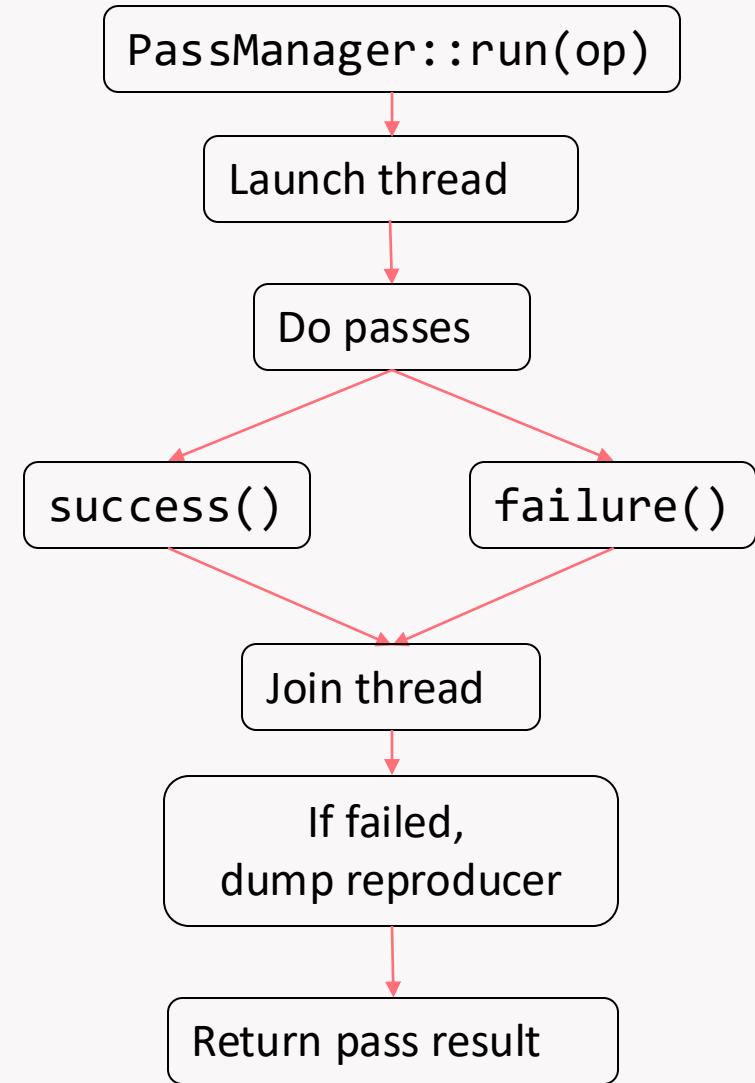
if [ $? -eq 0 ]; then
    exit 1
else
    exit 0
fi
```

```
$ mlir-reduce --reduction-tree='traversal-mode=0 test=interesting.sh'
reproducer.mlir
```

HOW DOES IT WORK?



Normal Run



Crash reproduction enabled

1. Pass options need to be *printable*

```
def ConvertVectorToLLVMPass : Pass<"convert-vector-to-llvm"> {  
    // ...  
  
    Option<"vectorTransformsOptions", "vector-transform-options",  
    "vector::VectorTransformsOptions",  
    "vector::VectorTransformsOptions()",  
    "...>,  
}
```

1. Pass options need to be *printable*



2. Be careful with thread-local state



```
thread_local SomeStorage x = ...;
```



```
auto stored = x.get();
```

PassManager::run(op)

Launch thread

Do passes

success()

failure()

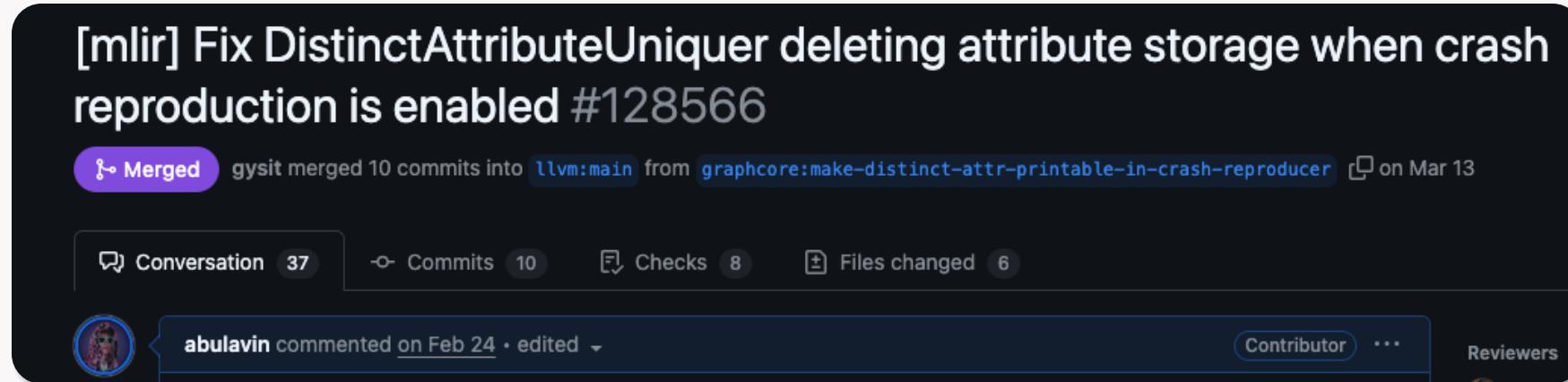
Join thread

If failed,
dump reproducer

Return pass result

Thread
lifetime

2. Avoid thread-local state



SUMMARY

Creating reproducers

Using mlir-opt and the PassManager

Integration

Using it in a real compiler

Using reproducers

Best practices for streamlining debugging

Internals

Everything you need to know to use
reproducers seamlessly



WHAT YOU CAN DO NEXT

1. Try the reproducers!

- Integrate them in your compilers
- Encourage your users to include them in bug reports

2. Add lit tests that exercise reproducers

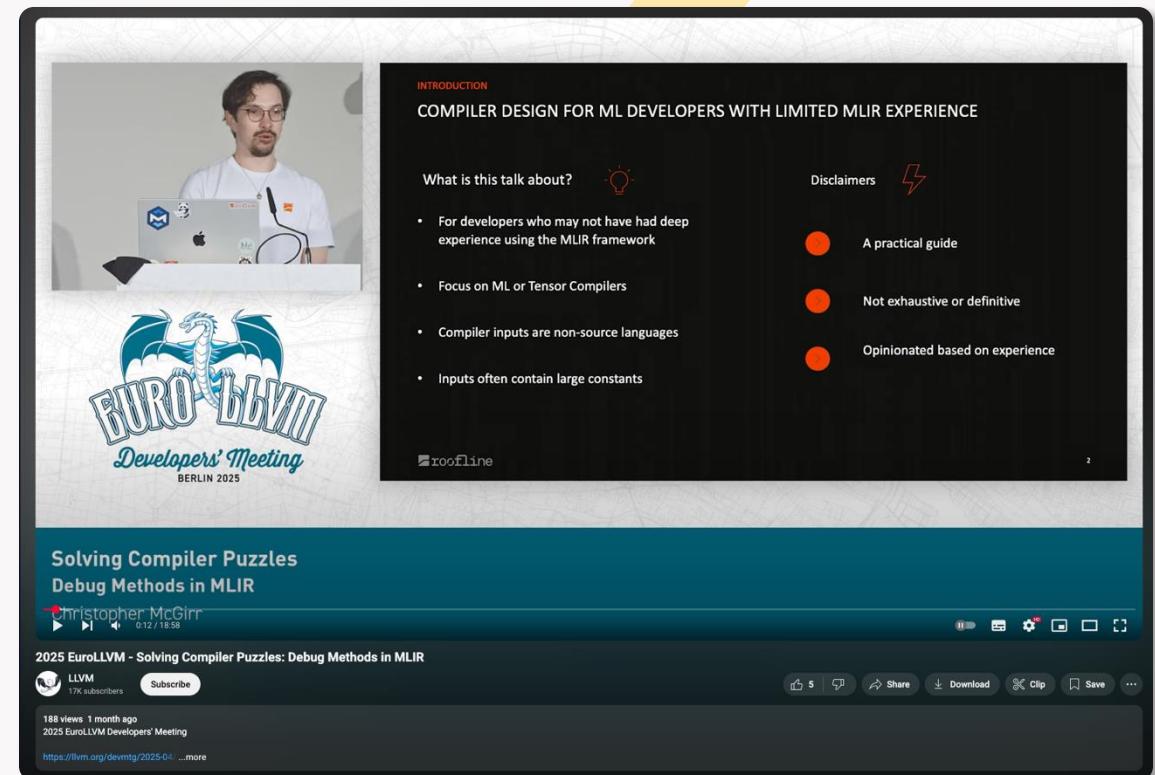
- Both upstream and in your projects

MORE RESOURCES

For more tools and tips on debugging MLIR:

2025 EuroLLVM - Solving Compiler Puzzles: Debug Methods in MLIR

Christopher McGirr



THANK YOU

Artemiy Bulavin

artemiyb@graphcore.ai

Github/Discourse: [@abulavin](#)



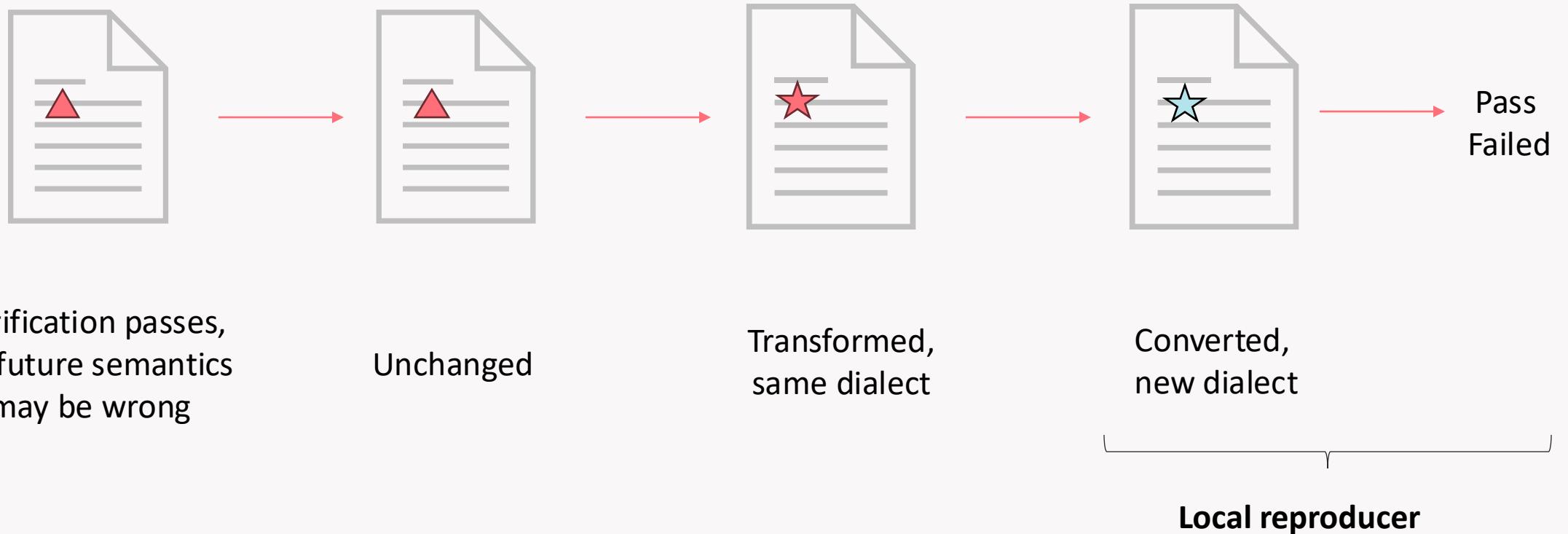
BONUS SLIDES



LOCAL VS FULL REPRODUCER

Local reproducers capture the state of the IR *just before failure...*

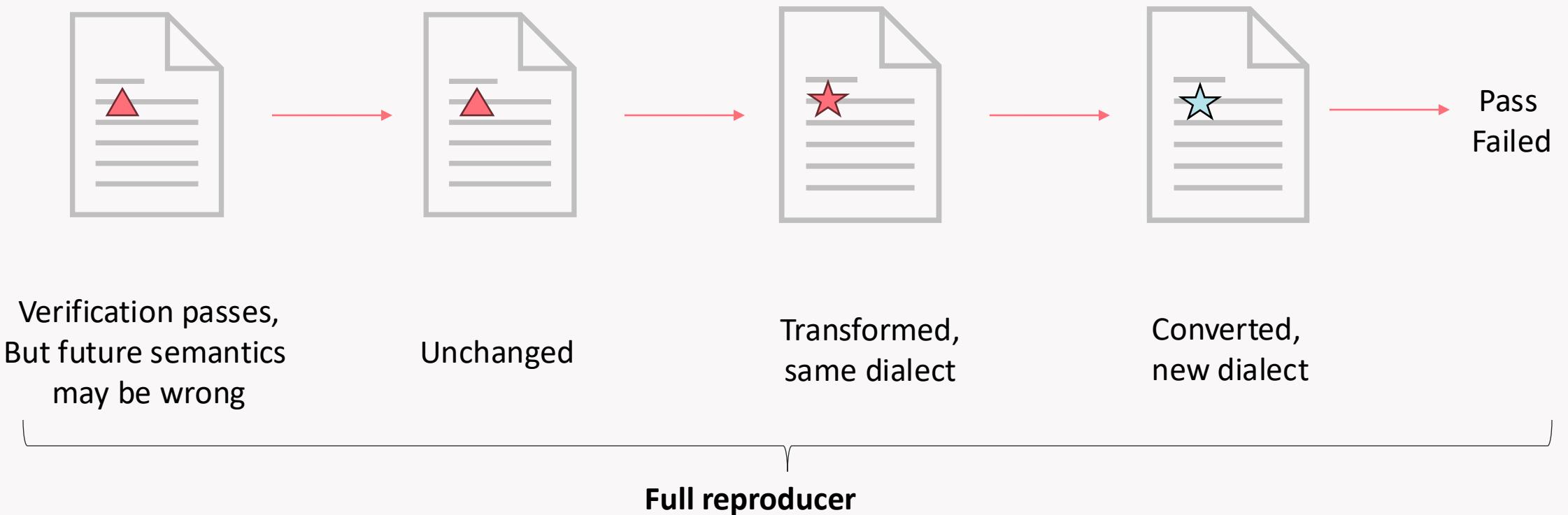
... but something bad may have happened much earlier



LOCAL VS FULL REPRODUCER

Local reproducers capture the state of the IR *just before failure...*

... but something bad may have happened much earlier



EXAMPLES OUT IN THE WILD

- **Triton**
- Tensorflow
- IREE

