

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

# Index

- $\bowtie$ , *see* relativization
- $-->$ , *see* *imp*
- $\vdash$ , *see* *derives*
- $\&\&$ , *see* *andp*
- address\_mapsto*, 372
- age*, 306, 340, 342, 343, 347, 348, 350
- age1*, 308
- age1\_join2*, 348
- Age\_alg*, 84, 347–349
- Age\_prod*, 350
- ageable*, 84, 299, 306, 343, 348, 349
- algNatDed*, 84, 86
- algSepLog*, 86
- aliasing*, 17
- ALL*, 52, 77
- allp*, 51, 77
- andp*, 51, 77, 342
- andp\_right*, 51, 342
- app\_mode*, 343
- app\_pred*, 308
- approx*, 368
- automation*, 60, 89–93, 125, 132, 136, 186–194, 201, 205, 411–441
- AV*, 352, 356
- AV.valid*, 367
- axiomatic semantics*, 5, 58, 119, 154
- axiomK*, 343
- backward proof*, 12
- big-step*, 26
- bisimulation*, 272
- boolean algebra*, 72
- box*, 343
- box*, 343
- C light*, 6
- Canc\_alg*, 39, 40, 54, 366
- cancel*, 194, 205, 206
- cancellative*, 39
- canonical form*, *see* *PROP/LO-CAL/SEP*
- certified compiler*, 3
- classical separation logic*, 82
- ClassicalSep*, 76
- clightgen*, 195, 205, 293
- coffee break*, 96, 106, 110, 297, 306
- comparison*, *pointer*, *see* *pointer comparison*
- CompCert*, 143, 146, 233–236, 385, 406
- AST*, 148
- expression evaluation*, 173–180

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

## INDEX

453

- front end, *see* `clightgen`
- memory, *see* `memory model`
- operational semantics, 288
- specification, 273–287
- completeness, 83, 88
- Concurrent separation logic, 69
- `concurrent-read`, 36, 70, 227
- `contravariant`, 64
- `core`, 38
- `core_duplicable`, 38
- `core_hom`, 38
- `core_idem`, 38
- `core_identity`, 39
- `core_self_join`, 38
- `core_unit`, 38
- `corec`, 66, 67, 123
- `corec_fold_unfold`, 66, 68
- `corec_least_fixedpoint`, 66
- `covariant`, 64
- `covariant`, 66
- `covariant_andp`, 67
- `covariant_const`, 67
- `covariant_const'`, 67
- `covariant_exp`, 67
- `covariant_id`, 67
- `covariant_orp`, 67
- `covariant_sepcon`, 67
- `Cross_alg`, 41
- `cross_split`, 41
- derives*, 50, 77, 340, 341
- `derives_cut`, 341
- `derives_trans`, 50
- `Disj_alg`, 40, 72, 366
- disjoint separation algebra, 40, 72
- disjointness, 40
- EK, *see* `exitkind`
- `emp`, 53, 54, 349
- `emp_sepcon`, 54
- `entailer`, 194
- `equiv_eq`, 50, 341
- `eval_expr`, 150, 162, 180
- `eward`, 81, 349
- `eward_sepcon`, 54
- EX, 52, 77
- `examples/cont`, 111
- `examples/cont/language.v`, 112
- `examples/cont/lifted_seplogic.v`, 136
- `examples/cont/lseg.v`, 124, 127
- `examples/cont/model.v`, 352, 358, 361
- `examples/cont/sample_prog.v`, 130
- `examples/cont/seplogic.v`, 117, 134, 136
- `examples/hoare/hoare.v`, 26, 30, 32
- `examples/lam_ref/`, 316
- `examples/sep/corec_example.v`, 66, 67
- `examples/sep/fo_seplogic.v`, 59
- `examples/sep/language.v`, 55, 57, 58
- `examples/sep/seplogic.v`, 59, 62
- `exitkind`, 395
- `exp`, 51, 77
- `exp_left`, 91
- `exp_right`, 91
- extensionality, 50, 58
- `extract-exists`, 23
- `fash`, 102
- `fash_triv`, 350
- FF, 51, 342
- `field_mapsto`, 184
- fixed point, 63

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

## INDEX

454

## Floyd

assignment rule, 12, 31, 161  
 Robert W., 8  
 VST automation system, 89, 208,  
 408

floyd/field\_mapsto.v, 184

floyd/loadstore\_lemmas.v, 185

footprint, 32

formal system, 76

forward, 189, 190, 192, 201, 204  
 of case study, 133

forward proof, 12, 415

forward simulation, 248, 273, 278,  
 279

forward-simulation, 282

frame inference, 189

frame rule, 18, 61

Freeable, 387

FUN, 355

funspec, 171

garbage collection, 82

global variable, 200

go\_lower, 191, 192, 200, 204

guard, 30, 31, 61, 116, 392, 394

heaplet, 17, 36

HORec, 109, 359

HOrec, 123, 124

HORec\_fold\_unfold, 125

identity, 53

identity, 38

imp, 51, 77, 342

Indir, 350

injection, 235, 248, 270, 282–284

injections, 247

intuitionistic separation logic, 82

IntuitionisticSep, 76

isolate, 23

join, 36

join, 37, 347, 348, 356

Join\_alg, 367

join\_assoc, 37

join\_canc, 39

join\_comm, 37

join\_core, 38

Join\_discrete adr, 59

join\_eq, 37

Join\_equiv, 59

join\_equiv, 350

Join\_fun, 60

Join\_lower, 59

join\_positivity, 37, 58

join\_self, 40

Join\_world, 60

knot, 296, 302–313, 325, 329, 357

later\_sepcon, 349

later\_wand, 349

laterM, 343

laterR, 343

level, 340, 368

list.v, 195

list\_dt.v, 212

listrep, 20

listrep, 124

lock, 227, 407

loeb, 120, 126, 344

Löb, Martin H., 344

logic, 76

%logic, 77, 83, 91, 186

loop invariant, 14

lseg, 214

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

## INDEX

455

- Lsh, 387
- magic wand, 54, *see* wand
- maps-to, 17, 60
- mapsto, 163
- memory model, 234, 235, 237–271, 289, 364
- modality, 342, 343
- model
  - CompCert memory, 234, 235, 237–271, 289, 364
  - concurrent separation logic, 230
  - of higher-order features, 340
  - of separation Hoare triple, 61
  - of separation logic, 84, 88
  - of Verifiable C logic, 362
  - share, 365
  - step-indexed, 84
- modified variables, 18, 61
- modus\_ponens, 51, 342
- modus\_wand, 54
- mpred, 134, 169, 407
- msl/ageable.v, 306, 307
- msl/alg\_seplog.v, 76, 85, 350, 359
- msl/alg\_seplog\_direct.v, 85
- msl/Axioms.v, 58
- msl/boolean\_alg.v, 72, 379
- msl/corec.v, 66
- msl/functors.v, 300, 301
- msl/knot.v, 310
- msl/knot\_full.v, 310
- msl/knot\_hered, 300
- msl/knot\_hered.v, 308–310
- msl/knot\_lemmas.v, 303
- msl/knot\_unique.v, 314
- msl/log\_normalize.v, 89
- msl/predicates\_hered.v, 308, 350
- msl/predicates\_rec.v, 359
- msl/predicates\_sa.v, 50
- msl/predicates\_sl.v, 85, 350
- msl/rmaps.v, 352, 357
- msl/sepalg.v, 43
- msl/sepalg.v, 37
- msl/sepalg\_generators, 60
- msl/sepalg\_generators.v, 45
- msl/seplog.v, 76
- msl/shares.v, 72, 380
- msl/subtypes.v, 99, 350
- msl/subtypes\_sl.v, 350
- msl/tree\_shares.v, 380
- mutex, *see* lock
- NatDed, 76, 77, 84, 102, 134
- natural deduction, 77, 134
- necessary, 342
- necR, 342, 343
- NO, 352
- no\_units, 42
- Nonempty, 387
- NoneP, 355
- normalize, 89
- now\_later, 344
- operational semantics, 26
- OracleKind, 154, 393
- orp, 51, 77, 342
- partial correctness, 10
- Perm\_alg, 37, 53, 54, 348
- permission algebra, 37, *see also* Perm\_alg, 53, 60
- permission share, *see* share
- permission, CompCert memory, 249–255, 262–263, 267–271, 286, 386–389

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,  
Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

## INDEX

456

- pointer comparison, 145, 149, 180–183, 246, 249, 252, 365
- Pos\_alg, 42, 59, 366
- positive permission algebra, 42
- positivity, 37
- pred, 295, 357
- %pred, 50, 52, 66, 77, 85, 87, 91, 308, 344
- pred\_hereditary, 348
- predicates in the heap, 98, 121, 230, 231, 355, 363, 364, 386, 388, 394, 407, 408
- predicates\_sl.v, 85
- preds, 356
- progs/list\_dt.v, 408
- progs/message.c, 217
- progs/verif\_message.v, 221
- progs/verif\_reverse.v, 198, 205
- prop, 52, 77, 344
- PROP/LOCAL/SEP, 186–205
- prop\_andp\_left, 92
- PURE, 352, 355
- pure, 21
- queue.c, 213, 408
- race, 69, 237, 250, 255
- Readable, 387
- rearrange, 23
- Reclndir, 100, 350
- recursive predicates, 63, 104
- relativization, 74
- resource, 33
- resource, 352, 367
- resource\_at, 356
- resource\_fmap, 356
- ret\_assert, 156, 159
- retval, 168, 198
- reverse.c, 195, 210, 408
- reverse.v, 195
- rewriting, *see* normalize
- rmap, 352, 356, 357, 367, 407
- Rsh, 387
- safety, 30
- same\_unit, 39
- sample\_prog.v, 133
- segment, 19
- semax, 61, 62, 141, 407, 408
  - of case study, 61, 62, 119, 130, 134
  - semantic model, 392–400
  - specific rules of, 154–172, 184, 185
- Sep\_alg, 38, 54
- SepAlg, 84
- separating conjunction, *see* sepcon
- separation algebra, 35
- separation logic, 3, 4, 16–24, 33, 35, 49, 55, 76–89, 116, 119, 130, 134, 142, 150, 153, 189, 226, 371, 393, 406, 412
  - classical, 82
  - concurrent, *see* concurrent separation logic
  - higher-order, 4, 54, 110, 111, 120, 347
  - intuitionistic, 82
  - lifted, 134–140
  - soundness, 26
- SeparationLogicSoundness.v, 400
- sepcomp, 287
- sepcon, 49, 81, 85, 89

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

## INDEX

457

- sepcon\_... , 53, 81, 82, 85, 87, 103, 348
- sepcon\_assoc, 53, 348
- sepcon\_comm, 53, 348
- sepcon\_cut, 54
- sepcon\_emp, 54
- SepInDir, 350
- SepLog, 76, 81, 84, 134
- SepRec, 350
- share, 36, 40, 41, 70–74, 163, 227, 228, 357, 364–366, 374–384, 386–387, 403
  - splittable, *see* split
- share, 72
- shared memory, 69, 222, 235, 237, 249, 250, 270, 272, 279, 365, 406
- Sing\_alg, 40
- small-step, 26
- soundness, 26, 58
- SoundSeparationLogic, 408
- split, 37, 41, 71, 74, 228, 365, 374–384, 386
- split\_core, 38
- split\_identity, 39
- squash, 298, 302, 313, 327, 356, 367, 403
- start\_function, 200
- static analysis, 4
- step index, 94–98, 105–110, 364
- step indexing, 64, 84, 97, 292, 316, 385, 392, 395
- StepInDir, 350
- stuck, 27
- subp, 102
- sumarray.c, 210, 211, 408
- synchronization, 36, 69, 223–227, 275, 404
- tactic, 89, 92, 194, 200–206
- tc\_expr, 173–183
- tc\_assert, 175
- tc\_expr, 157, 159, 160, 166
- the\_unit, 40
- thread, 36, 156, 222–232, 249, 255, 276, 289, 364, 375, 386, 392, 402, 407
- token factory, 40, 71, 377
- Triv, 100, 102, 350
- Tsh, 69, 387
- TT, 51, 342
- type context, 178, 180, 201
- typecheck, 118, 157, 164, 166, 173–183, 201
- typed\_mapsto, 208
- typed\_mapsto\_ , 208
- unfash, 350
- unit\_core, 39
- unit\_for, 38
- unit\_identity, 39
- unsquash, 356, 367
- valid, *see* AV.valid
- veric/binop\_lemmas.v, 173
- veric/Clight\_new.v, 276, 288
- veric/envirion\_lemmas.v, 173
- veric/expr.v, 150, 173, 408
- veric/expr\_lemmas.v, 173
- veric/ghost.v, 394
- veric/juicy\_ext\_spec.v, 391
- veric/juicy\_mem.v, 386, 388
- veric/lift.v, 138
- veric/res\_predicates.v, 371

Cambridge University Press

978-1-107-04801-0 - Program Logics for Certified Compilers

Andrew W. Appel, Robert Dockins, Aquinas Hobor, Lennart Beringer, Josiah Dodds,

Gordon Stewart, Sandrine Blazy and Xavier Leroy

Index

[More information](#)

---

INDEX

458

veric/rmaps.v, 367  
veric/semax.v, 392, 398  
veric/semax\_call.v, 398  
veric/semax\_lemmas.v, 399  
veric/semax\_loop.v, 398  
veric/semax\_straight.v, 398  
veric/SeparationLogic.v, 408  
veric/SequentialClight.v, 400  
vst/progs, 210  
Vundef, 240  
  
wand, 81, 349  
wand\_sepcon\_adjoint, 54  
world, 60  
Writable, 387  
  
YES, 352, 355