

Week 8: Python Pipeline Fix - Quick Summary

Status:  COMPLETE

Date: January 31, 2026

Commit: d808321

What Was Done







Fixes Applied

1. **Fixed Python spec_to_ir.py** - Corrected IR step format (`kind` → `op`)
2. **Fixed Python ir_to_code.py** - Added `byte[]` type mapping
3. **Validated End-to-End** - Tested with real specs (ardupilot_test)



Results

-  Python IR now matches Rust IR format exactly
-  Generated C code is valid with correct types
-  Multi-file spec merging works
-  All code generation targets tested (C, Python, Rust)



Documentation Created

- `docs/WEEK8_PYTHON_IR_INVESTIGATION.md` - Root cause analysis
 - `docs/WEEK8_COMPLETION_REPORT.md` - Full completion report
 - `test_outputs/PIPELINE_COMPARISON.md` - Pipeline comparison
-

Quick Test

```
# Generate IR
python3 tools/spec_to_ir.py \
  --spec-root spec/ardupilot_test \
  --out test_outputs/python_pipeline/ir.json \
  --lockfile local_toolchain.lock.json

# Generate C code
python3 tools/ir_to_code.py \
  --ir test_outputs/python_pipeline/ir.json \
  --lang c \
  --templates templates/c \
  --out test_outputs/python_pipeline/
```

Key Metrics

Metric	Before	After
Functional Pipelines	2/4 (50%)	3/4 (75%)
Python IR Format	✗ Wrong	✓ Correct
Code Generation	✗ Failed	✓ Success
Multi-File Support	⚠ Partial	✓ Full

Files Changed

- `tools/spec_to_ir.py` (3 changes)
- `tools/ir_to_code.py` (1 change)
- Documentation (3 new files)
- Test outputs (validated)

Next Steps (Week 9)

1. Add multi-file spec support to Rust/SPARK
2. Improve SPARK IR generation quality
3. Begin Haskell pipeline work

For full details, see: `docs/WEEK8_COMPLETION_REPORT.md`