R53 Multiple Assigner DFDs

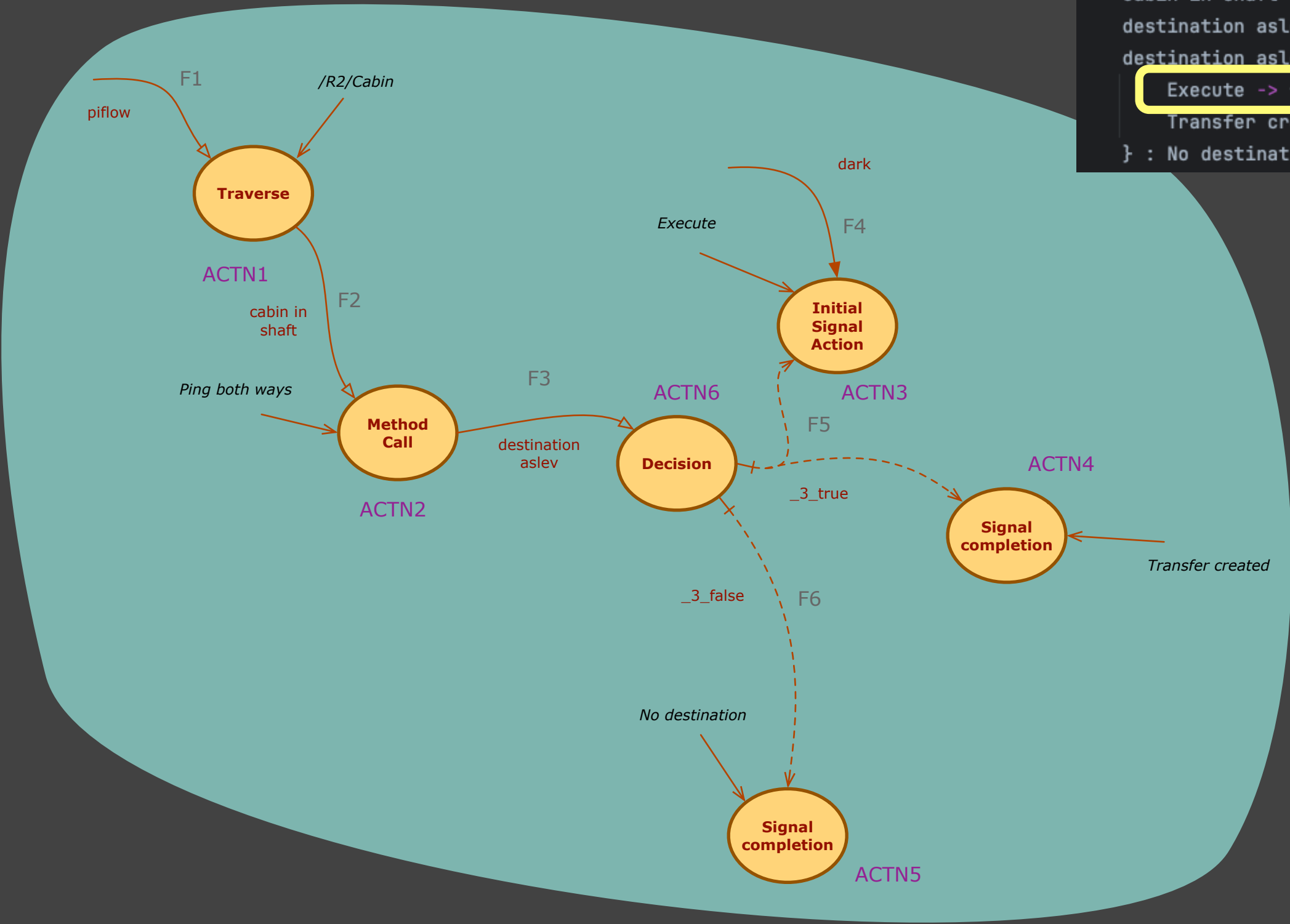Leon Starr
2025-10-12/ v1.3

A23

NO TRANSFER

// Waiting for service request

No actions

A25

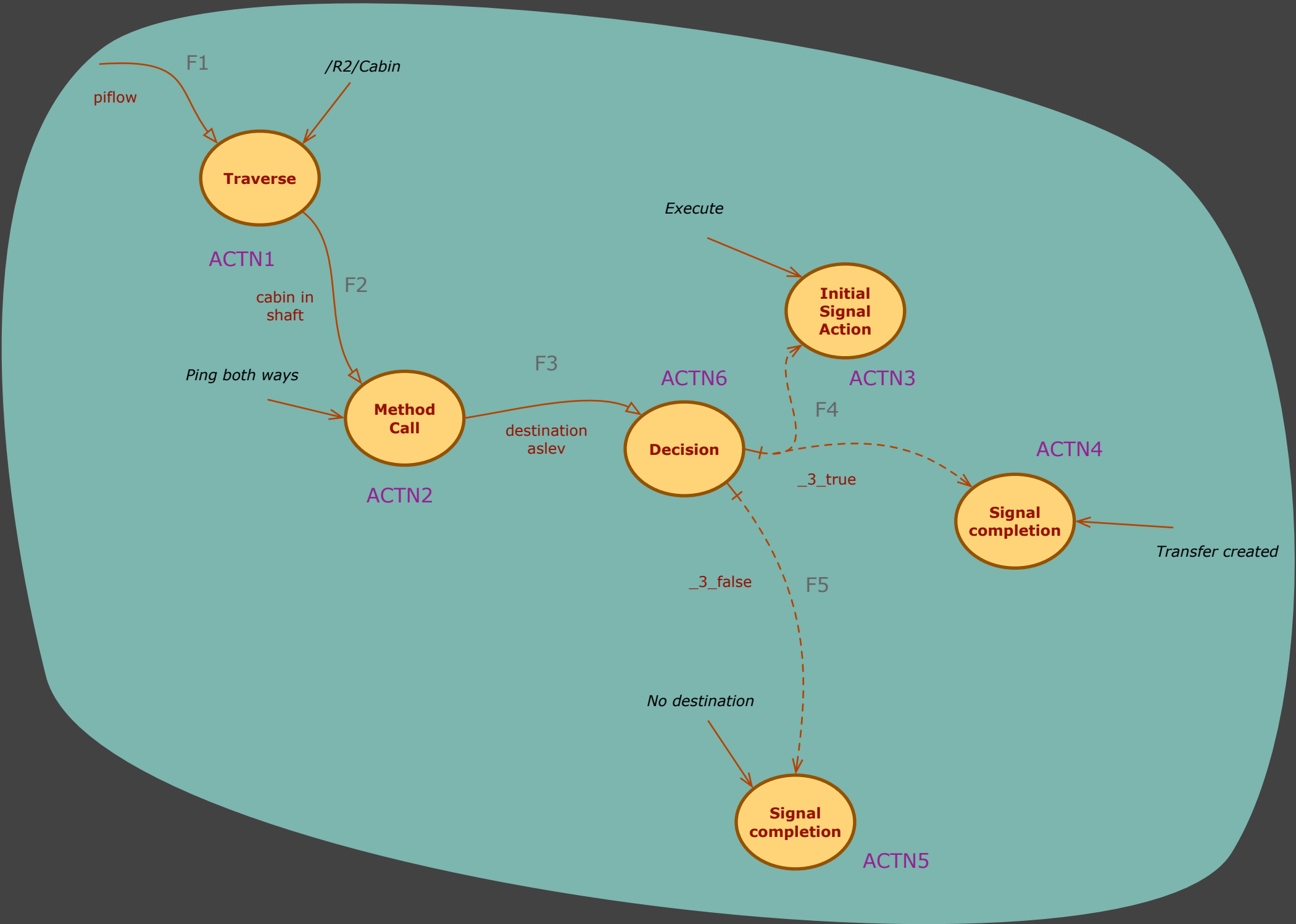Added Tint/Color attributes to test setting of explicit and default scalar attribute values

```
state Search for new destination
activity
    // The Cabin in this Shaft is stationary and has no Transfer
    cabin in shaft .= /R2/Cabin
    destination aslev .= cabin in shaft.Ping both ways()
    destination aslev? {
        Execute -> *Transfer(Tint: _dark) &R53 cabin in shaft, destination aslev
        Transfer created -> me
    } : No destination -> me
```

F1

/R2/Cabin

piflow

Traverse

ACTN1

F2

cabin in
shaft

Ping both ways

**Method
Call**

ACTN2

F3

destination
aslev

Execute

dark

F4

**Initial
Signal
Action**

ACTN3

ACTN6

Decision

F5

_3_true

ACTN4

**Signal
completion**

Transfer created

_3_false

F6

No destination

**Signal
completion**

ACTN5

Search for new destination

// The Cabin in this Shaft is stationary and has no Transfer
cabin in shaft .= /R2/Cabin
destination aslev .= cabin in shaft.Ping both ways()
destination aslev? {
    Execute -> *Transfer &R53 cabin in shaft, destination aslev
    Transfer created -> me
} : No destination -> me

A25

piflow
F1
/R2/Cabin

Traverse

ACTN1

F2
cabin in
shaft

Execute

**Initial
Signal
Action**

ACTN6
ACTN3

Ping both ways

**Method
Call**

F3
destination
aslev

Decision

F4

ACTN4

ACTN2

_3_true

**Signal
completion**

Transfer created

_3_false
F5

No destination

**Signal
completion**

ACTN5

Search for alternate destination

A27

```
activity
    cabin in shaft .= /R2/Cabin
    fwd = cabin in shaft.Travel direction
    xfer .= cabin in shaft/R53/Transfer
    call in progress .= xfer/R53/Accessible Shaft Level/R49/Floor Service( Direction: fwd )

5   fwd dest .= cabin in shaft.Ping( dir: fwd )
6   fwd dest? dest aslev .= fwd dest :
        !call in progress? {
            // Search opposite the travel direction only if not servicing
            // a floor request in the current direction
            rev = fwd.opposite
            rev dest .= cabin in shaft.Ping( dir: rev )
            rev dest? cabin in shaft.Travel direction = rev
            // may be empty
            dest aslev .= rev dest
        }

7   dest aslev? {
        dest floor = dest aslev.Floor
        dest floor != xfer.Destination floor?
            Dest change request( new dest: dest floor ) -> xfer
    }

8   Dest change evaluated -> me
```

Panel 1 (ACTN1–ACTN5):

- F1 · /R2/Cabin · _partition_instance
- Traverse — ACTN1
- cabin in shaft · F2 · /R53/Transfer
- Traverse — ACTN3
- Travel direction
- Read — ACTN2
- xfer · F4 · /R49/Floor Service
- Traverse — ACTN4
- fwd · F3
- Floor Service.Direction · ALL
- Restriction Condition
- F5
- Select — ACTN5
- Floor Service.Direction == fwd
- F6 · call in progress

Panel 2 (ACTN6–ACTN16):

- F2 · cabin in shaft
- F3 · fwd · Ping
- Method Call ⑤ — ACTN6
- fwd dest · F7
- Decision ⑥ — ACTN16
- Pass — ACTN7
- F16 · _16_true
- F6 · call in progress — ACTN8
- Computation
- F17 · _16_false
- F9 · _NOT <F6>
- Decision — ACTN15
- not call in progress
- F8 · _7_dest aslev
- F15 · _15_true
- dest aslev · F14 — ACTN14
- F3 · ACTN9
- fwd
- Type Operation
- cabin in shaft · F2 · Ping
- opposite · rev
- F13 · _13_dest aslev
- F10
- Method Call — ACTN10
- Pass — ACTN13
- F11
- rev dest
- Decision — ACTN12
- _12_true · F12
- Write — ACTN11
- Travel direction
- cabin in shaft · F2
- CABIN

Panel 3 (ACTN17–ACTN23):

- F14 · dest aslev
- Decision — ACTN22
- F4 · xfer
- Destination floor
- Read — ACTN18
- F22 · _22_true
- F19
- dest floor != xfer.Destination floor
- Read — ACTN17
- Floor
- dest floor
- F18
- Computation — ACTN19
- _<F20> != <F21> · F20
- Decision — ACTN21
- Destination change request
- Signal instance set — ACTN20
- F21 · _21_true
- xfer · F4
- Dest change evaluated
- Signal completion — ACTN23