ESP Fluid Tracking

5/29/22 Brent Roman brent@mbari.org

What is Fluid Tracking?

- The 2G ESP can follow fluid movements
  - between Syringes and containers
  - into and out of clamped pucks
  - into and out of external Intake and Exhaust
- Fluid tracking works in real and simulated modes
  - can be configured to stop operation
    - just before a reagent bag is “pulled flat”
- Fluid tracking slows simulation by about 20%

Using Fluid Tracking

```
-> require 'fluid'  #enables fluid tracking
-> Tank.clear #empties all virtual containers
```

- Run ESP normally... later...

```
-> Tank  #displays long table summarizing fluid use
#Fluid::Vessel#  Net Volume  (Lowest Volume)
```

- Fluid::Vessel = name of the fluid container
- Net Volume = change in volume of fluid since `Tank.clear`
- Lowest Volume = least volume remaining in container
  - only tracked for containers where fluid is pumped both ways
- first column of each line is a special code
  - '=' denotes closed fluid container (eg. :flush)
  - '~' denotes container that is vented to atmosphere (eg. ~waste)
  - ':' denotes fluid added to or removed from above container

Limiting Reagents' Supply

```
-> require 'fluid'      #enables fluid tracking
-> Tank.clear      #empty all virtual containers
-> Tank.supply flushCSR:12, flashPSR:5
        #supply only 12ml flushCSR, 50ml flushPSR
-> bac 100      #start the usual bac protocol
...
Pulling 5ml of CSR.flush
CSR.select :flush
CS.seek 5ml
Fluid::Depleted in simfast -- flushCSR pulled dry by Collection Syringe
-> Tank.supply flushCSR:20   #hang a new virtual flush bag
-> Thread.resume   #resume from Fluid::Depleted error
```

Banking Virtual Reagents

```
-> require 'fluid'      #enables fluid tracking
-> Tank.clear          #empty all virtual containers
-> Tank.bank flushCSR:12, flashPSR:5
        #deposit balance of 12ml flushCSR, 50ml flushPSR
-> bac 100             #start the usual bac protocol
...
```

- Banked reagents are not depleted when they run out
  - instead, their balance simply goes negative!
- If you use Tank.supply and Tank.bank for different reagents
  - only those you last "supplied" can raise Fluid::Depleted errs
- Tank.clear zeros the balance of all reagents

Logging Fluid Movement

```
-> require 'fluid'  #enables fluid tracking
-> Tank.clear      #empty all virtual containers
-> Fluid.logTransfer true   #log fluid movements
-> bac 100 #start the usual bac protocol
```

... Log excerpted during Puck Evac ...

```
<simfast> CBV.dial SSV
@23:29:16.16 <connect> CC-->SS 3.341ml @ 3.01bar
CS-->SS 0.792ml @ 3.58bar
<simfast> CTV.dial :air,avoiding: :intake
@23:29:17.66 CC-->air 10.072ml
SS-->CC 2.817ml @ 2.29bar
SS-->air 6.454ml
...
```