Drunc status

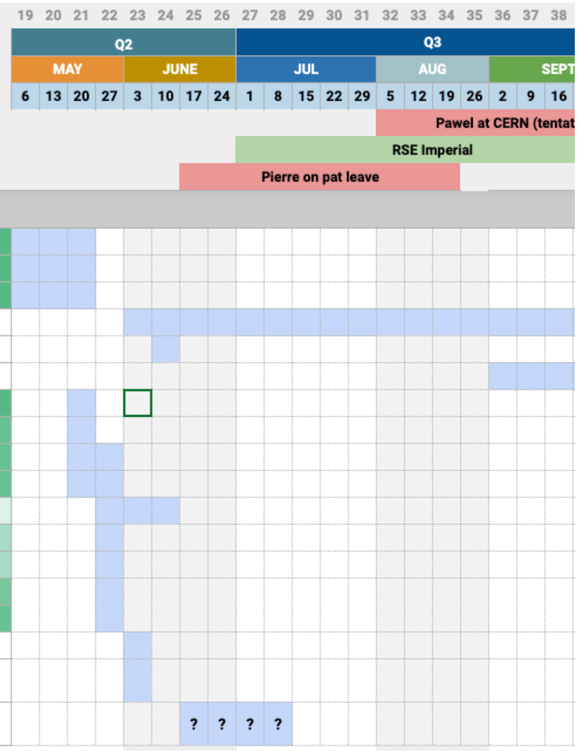
Pierre (a lot work being done by Pawel & Tiago)

Introduction

- Baseline plan: cut v5.1.0 on the 12th July, as a basis for NP-VD running
  - Use drunc / OKS
  - Release Gantt chart: https://cern-my.sharepoint.com/:x:/g/personal/giovanna_lehmann_cern_ch/EdjPUXsxU6VLmAZ1olEgDVoBNrjojTqg-aB2OY8Ydzl2-A?rtime=mWhgs95_3Eg
- This means drunc needs to be in a usable state for v5.1.0
- I'm on paternity leave from the 17th June (until the 26th August)
- Planned the development before I leave and need to make sure people can develop on drunc while I'm away.

Gantt chart

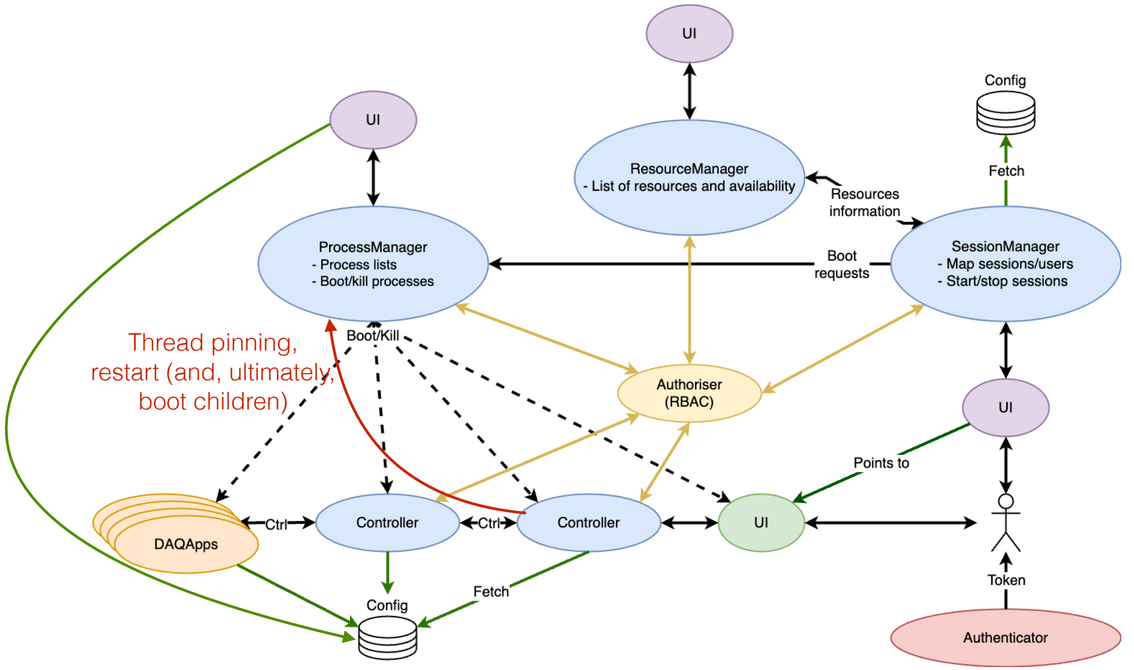
WBS NUMBER	TASK TITLE	WHO	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	**Drunc controller core development**					
1.1	Rationalisation children error modes	Pierre + Pawel	6/5/24	27/5/24	21	100%
1.1.1	Modification of druncschema	Pierre + Pawel	6/5/24	27/5/24	21	100%
1.1.2	Core changes	Pierre + Pawel	6/5/24	27/5/24	21	100%
1.2	FSM sequences	Pierre + Pawel	27/5/24	30/9/24	126	0%
1.2.1	Client side FSM sequence (if time allows!)	Pierre + Pawel	27/5/24	31/5/24	4	0%
1.2.2	Server side FSM sequence	Pierre + Pawel	2/9/24	30/9/24	28	0%
1.3	Rename FSM interfaces to FSM action	Pierre + Pawel	20/5/24	5/24/2024	4	100%
1.4	FSM information passthrough (child object to get some data from parent I	Pierre + Pawel	27/5/24	3/6/24	7	90%
1.4.1	Core changes	Pierre + Pawel	27/5/24	24/6/24	28	90%
1.4.2	Basic FSM pre/post transition actions	Pierre + Pawel	3/6/24	24/6/24	21	90%
1.6	Documentation (wiki)	Pierre + Pawel	10/6/24	15/6/24	5	20%
1.7	Application discovery service	Pierre + Pawel	23/6/24	31/6/24	#VALUE!	50%
1.7.1	cmdlib (trivial), restcmd (non-trivial) changes	Pierre + Pawel	23/6/24	31/6/24	#VALUE!	50%
1.7.2	microservice (may not need to be)	Pierre + Pawel	23/6/24	31/6/24	#VALUE!	80%
1.7.3	drunc-controller	Pierre + Pawel	23/6/24	31/6/24	#VALUE!	90%
1.8	Thread pinning	Pierre + Pawel	10/6/24	15/6/24	5	0%
1.8.1	Thread pinning placeholder (json-based affinity script?)	Tiago + Pawel + Gordon	10/6/24	17/6/24	7	0%
1.8.2	Thread pinning through OKS	Tiago + Pawel + Gordon	17/6/24	15/7/24	28	0%



- Live version is here and being updated as we go along:
  - https://docs.google.com/spreadsheets/d/1cdFtOPz-8KkDiNY7ZYbTIzWP3Z9NQdi3bNietXOjFCo/edit#gid=1115838130

Status update

- Rationalisation of error mode
  - Done, now each FSM transition gives a status of the transition for all the descendents
- FSM information passthrough: Done
- FSM objects renaming: Done, FSMInterface → FSMAction
- FSM actions (“any action done before or after a transition”)
  - Core changes to enable them: Done (very few changes needed)
  - Examples: Done (making the actions themselves do something interesting and appdal changes)
- Thread pinning
  - Hasn’t started, needs coredal changes and a meeting to understand what exactly we need to do
  - Where should the thread pinning json file go?
  - Who should execute this?
    - Process manager knows where things are running, Controller knows about FSM commands
      → Controller talks to the PM and tells it to execute the thread pinning, how does the controller knows where the process manager is?



- Application discovery service (extra, aka no more port offseting)
  - Reason this is here is because the Process manager could advertise itself on the app discovery service
  - Some work already done by me a couple of weeks (months?) ago, we got a test system more or less working last week with Pawel.
  - A bit worried this may be scope creep, should we go ahead with this?
  - We don’t need recreate the app discovery service, and use the connectivity service
  - We then have another problem: session names, Session Id are used to get the configuration from the OKS DB, this means the session name is hardcoded in the configuration. Pretty sure we will end up stepping on each others toes quickly (with the connectivty service).

Status update

- Real run archiving facilities
  - Run number: Tiago showed me an example of this working a minute ago
  - Run registry: Complexity lies in putting *all* the configuration into the run registry. It's hard to parse <includes> in the xml. - Ask Eric.
  - ELisA
    - microservice: in contact with Jonathan to resurrect the ELisA microservice he had created a year ago
    - FSM action: Jonathan
- Drunctiming
  - Need to discuss with Stoyan
- Other misc
  - Wiki update (before RSE come to help)
  - Testing facilities ← Something that keeps bitting us with nanorc. I would like to make drunc development test-driven, and we rely on github actions to trigger a CI every time we PR. We're already quite late in the development, which is a bit of a shame…

Summary

- Quite a bit of development over the last weeks and in the coming weeks
- Feeling a bit more optimistic than a few weeks ago to get drunc ready for v5.1.0