



U.S. DEPARTMENT
of ENERGY

EIC-BeamAI

Controls Level Digital Twins

Levente Hajdu

September 2, 2026

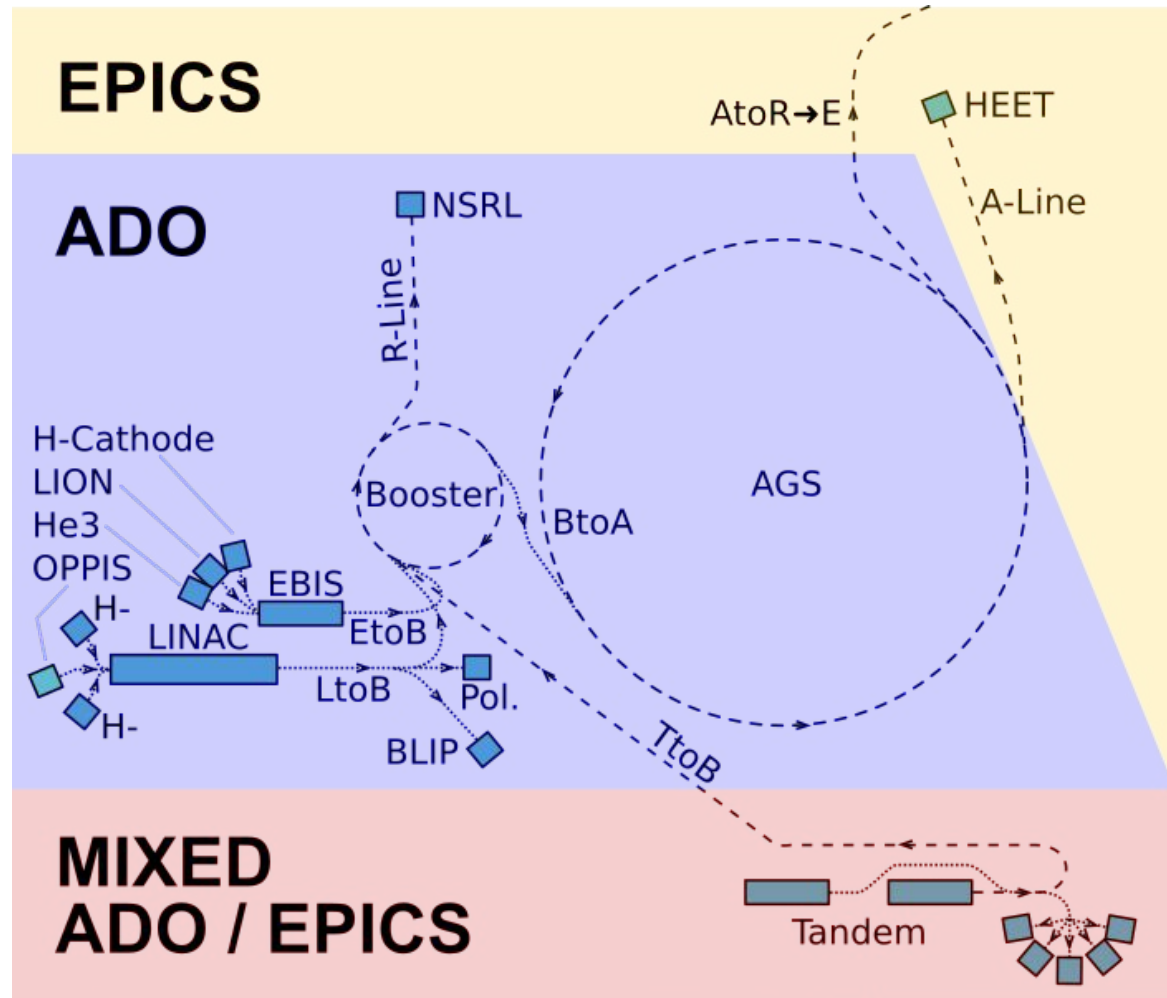
X f i n @BrookhavenLab

Introduction

- Facility, Control System(s), and approximate timeline for operations
- Benchmarks with various beamline simulators
- Basic control system connected model
- Design questions for Genasis deliverables

State of the Control - System

- The bulk of the existing control system is ADO (Accelerator Device Objects)
- New parts (EIC and some Tandem upgrades) will use EPICS
- There is a desire to upgrade older parts of the control system to EPICS, but as yet no plan or funding exists, and the system is complex (time-based synchronized functions and pulse multiplexing)



Accelerator Device Objects (ADO's) and Multinet The Basics

```
Terminal
File Edit View Search Terminal Help
acnlinmf 141:adoIf -vo -4 simple.test shorts
5
acnlinmf 142:adoIf -4 simple.test shorts 10
adoIf simple.test shorts 5
adoIf simple.test shorts 10
acnlinmf 143:adoPet simple.test &
```

/tmp/simple.test.ado PPM User: RHIC_U4

Page PPM Device Data Tools Buffer Help

ado class simple

simple.test	fecName	acnlin6a.pbn.bn1
simple.test	description	Examples of para
simple.test	constructTime	1759349650
simple.test	version	1.8 / Jan 9 202
simple.test	className	simple
simple.test	commandBuffer	%
simple.test	commandBufferLength	0
simple.test	testC	0
simple.test	commandC	commandC
simple.test	menuS	first
simple.test	charS	
simple.test	ucharS	
simple.test	shortS	5
simple.test	ushortS	5
simple.test	intS	5
simple.test	dintS	0
simple.test	longS	1
simple.test	ulongS	1
simple.test	floatS	0
simple.test	doubleS	0

(19,3) simple.test:ushortS Nudge: 1 255

Thu Oct 16 18:39:45 2025: UpdateCellDisplay (40,3) - Unable to extract data
Thu Oct 16 18:39:45 2025: Get and Async requests complete.

PpmUserMon

Setup Diagnostics Help

Tue Oct 14 16:25:21 SC Length 5.8s

Super					F	
AGS					7	
Booster				N6	A7	
Tandem	A1					
# cycles	9					
Linac				6		
ETB				N6		
EBIS				N6		
ion type				Si		
# cycles				1		

Server Room

Server Node
(physical computer)

Manager
(Server Process
(ADO Instance))

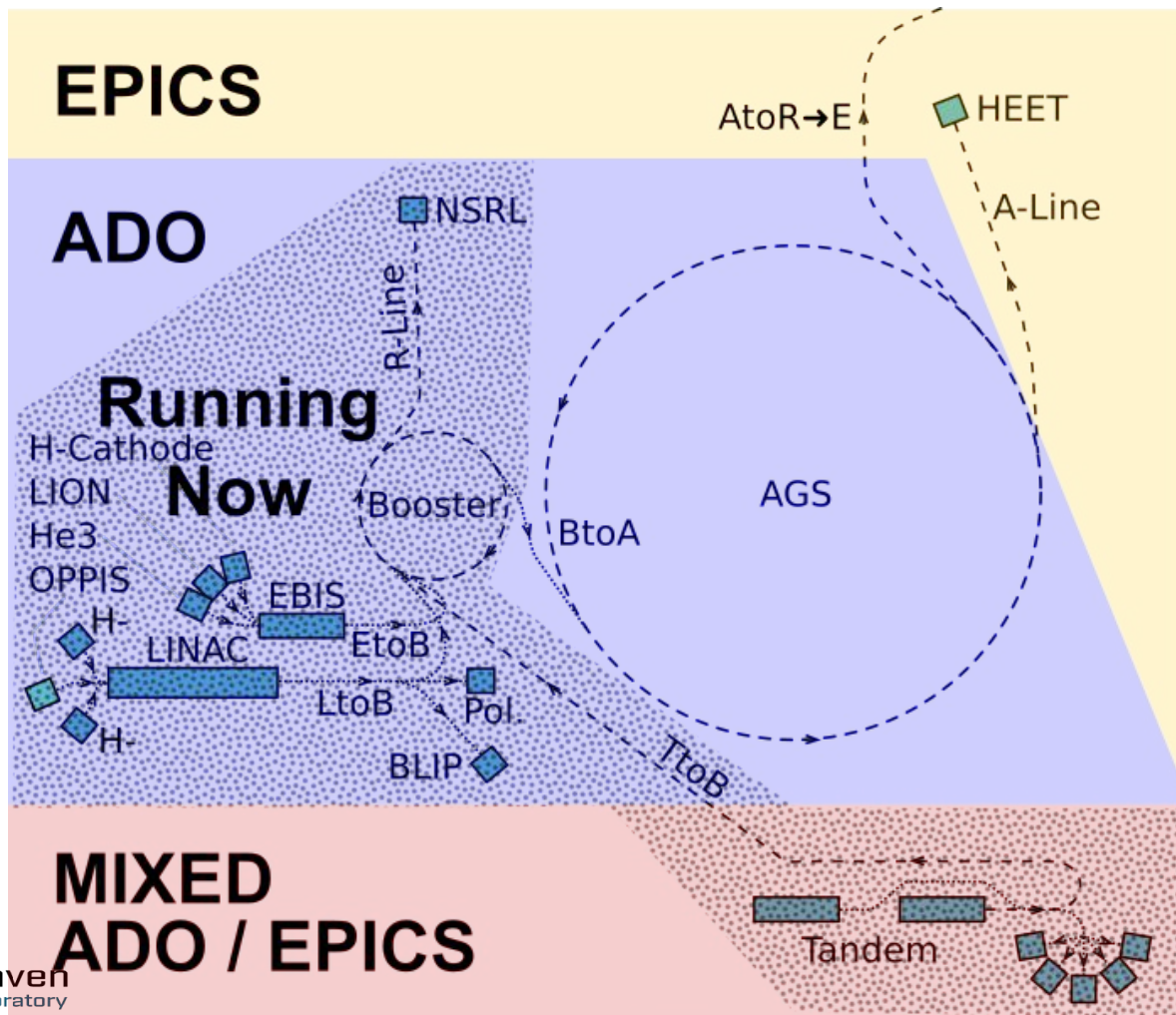
Process
Variable

Variable [8]

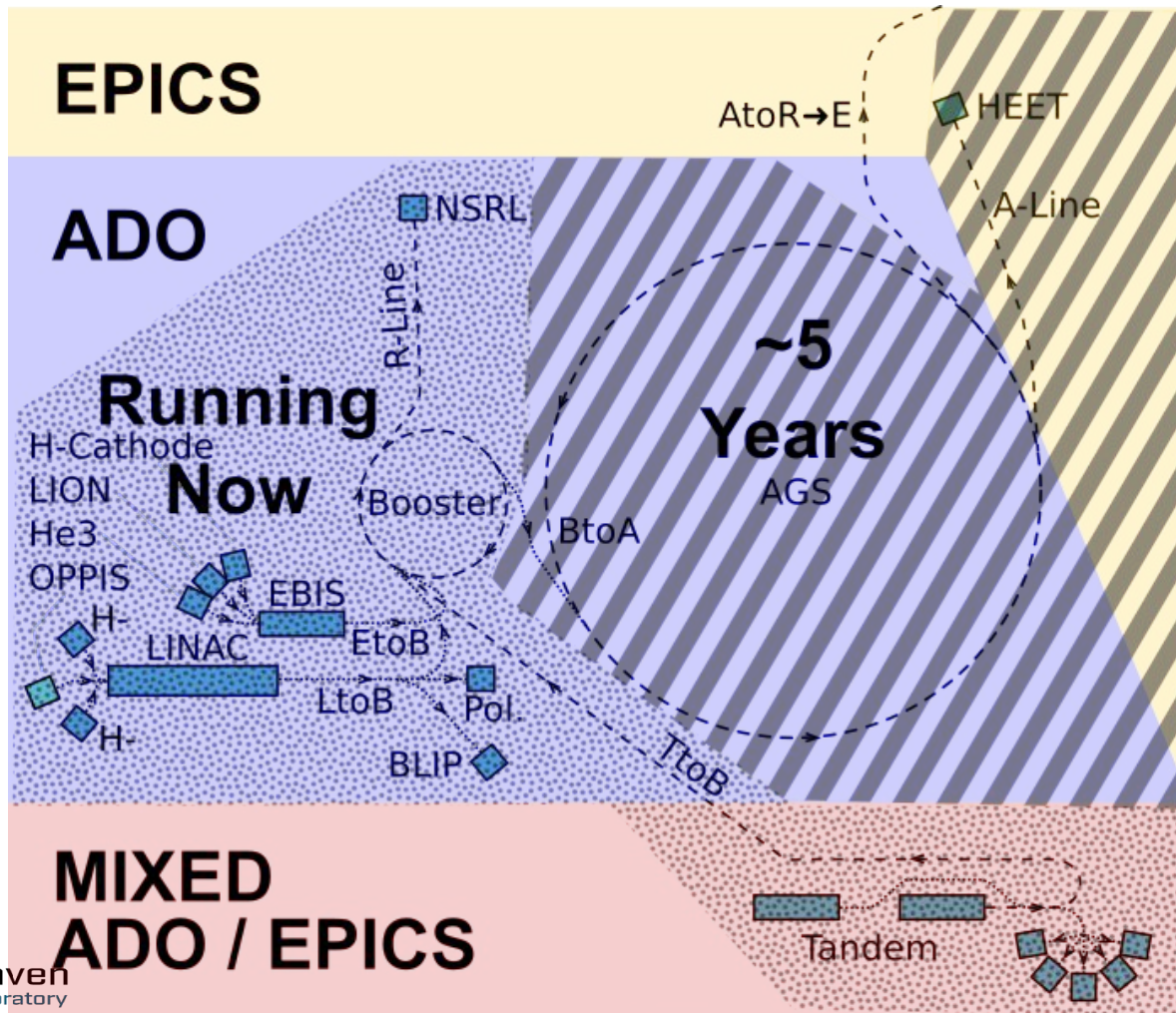
Setter Code

Manager

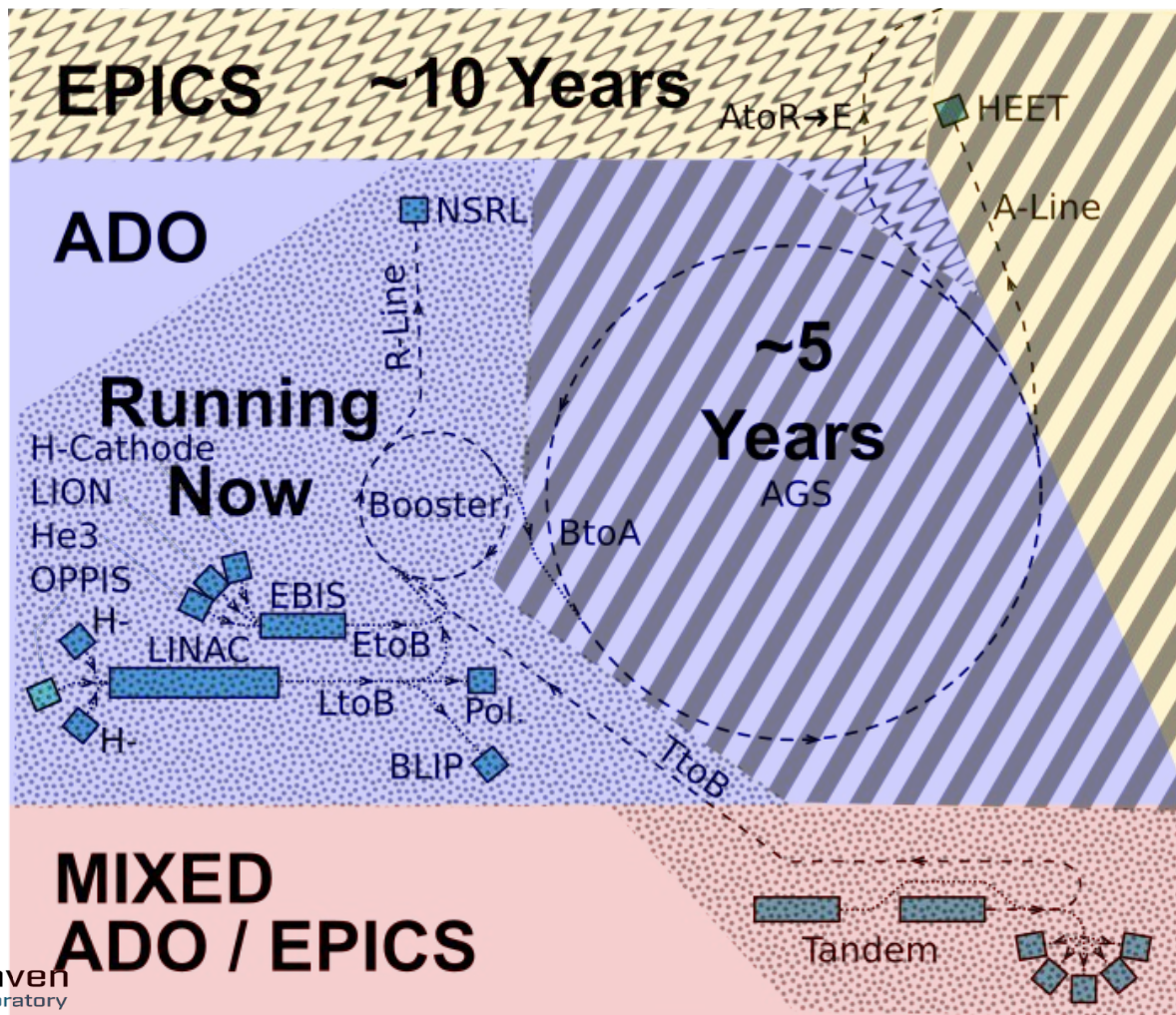
Approximate
Timeline for
Facility
Operations



Approximate Timeline for Facility Operations



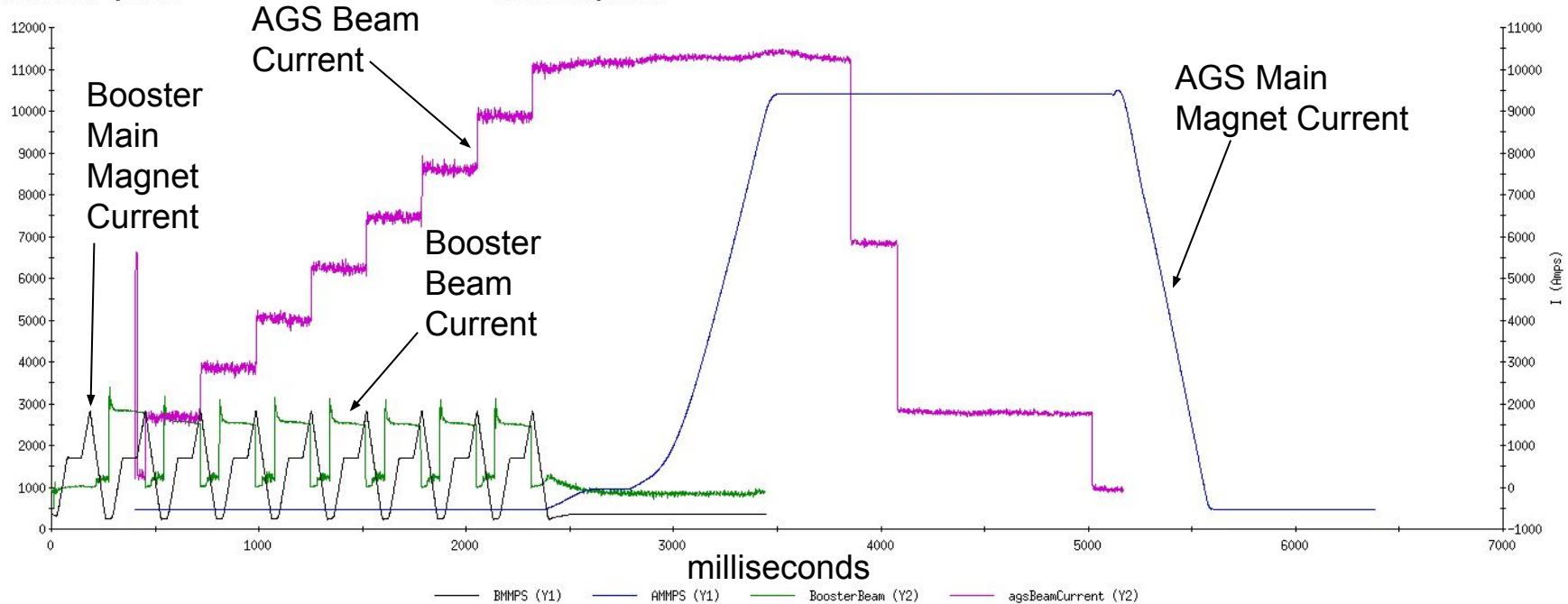
Approximate
Timeline for
Facility
Operations



Facility Operation During a RHIC Fill

10:03:42 2025, cycle 35934

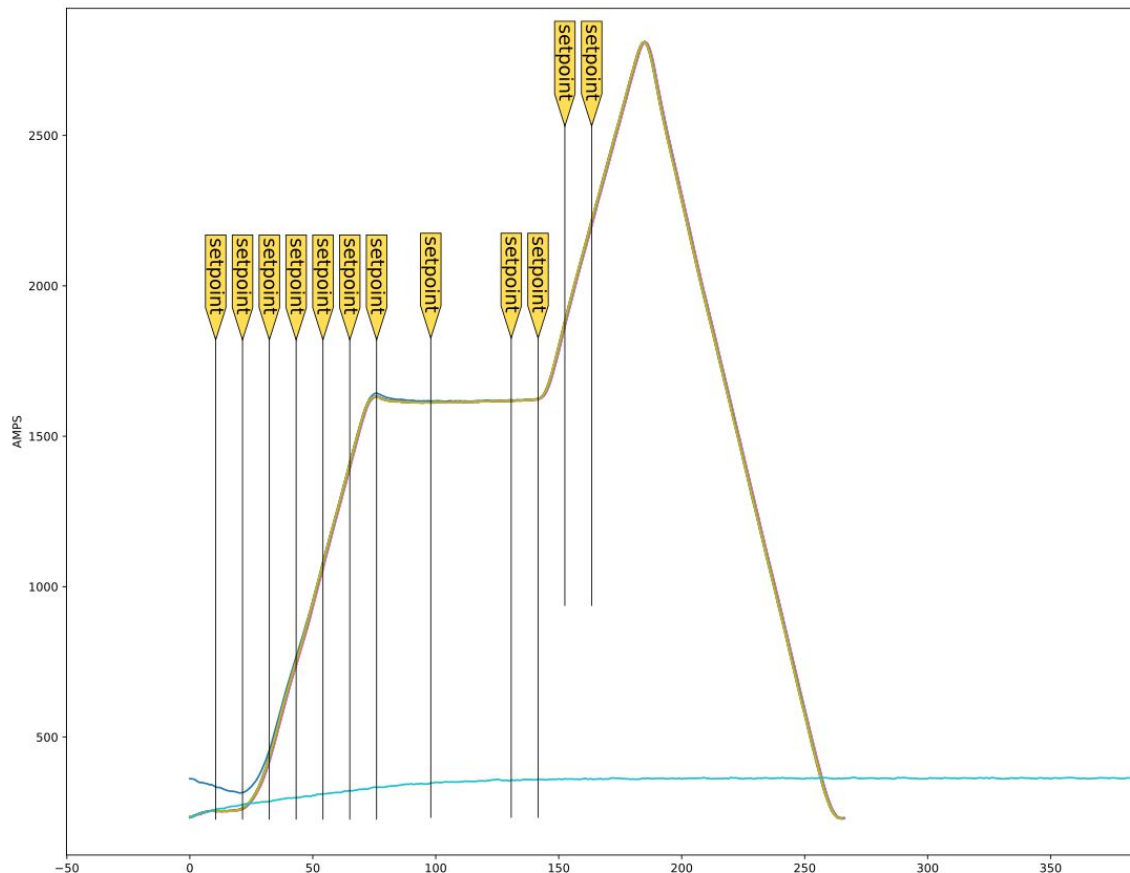
test Booster/Ags current



Digital Twin Modelling

Do we break into different models with discrete setpoints or feed in time-based functions?

- Time needed to run
- Effects such as beam pipe-induced current need time-based functions
- Up until now, I had been working on the assumption of discrete points/models



Digital Twin Modelling

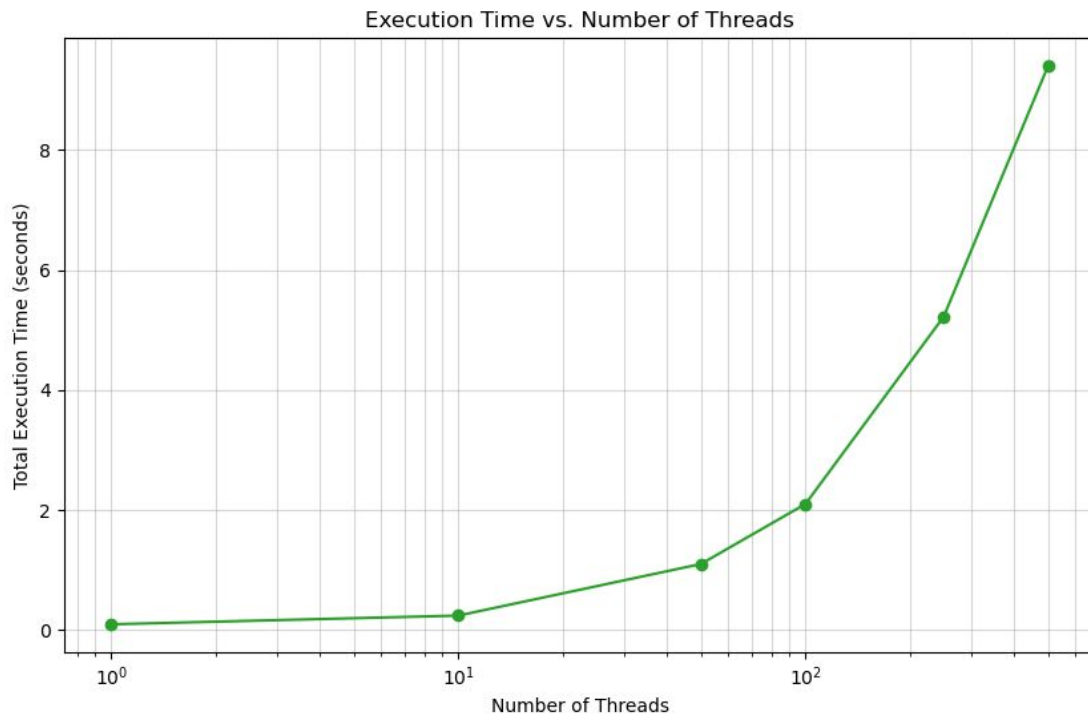
- ADOs only support 1D arrays. Going down the time-slicing route, we would do something similar to our BPM ADO where you enter times and get back positions. We would enter times and return concatenated Beta and Orbit arrays.

Input = [T1, T2, T3]

Output = [
 $\beta_{T1[0]}, \beta_{T1[1]}, \beta_{T1[2]}, \dots,$
 $\beta_{T2[0]}, \beta_{T2[1]}, \beta_{T2[2]}, \dots,$
 $\beta_{T3[0]}, \beta_{T3[1]}, \beta_{T3[2]}, \dots]$

Madx Benchmarks

I have tested Madx with different numbers of threads [1, 10, 50, 100, 250, 500, 750, 1000]. It slowed down my computer substantially, so I could not really run anything else while the benchmark was running. At best, the computer could do about 7 threads at one time. 1K threads take 120 seconds. The function we tested is Lucy's BumpGUI lattice (fully buffered string in memory, including called subfiles), with one value changed in each iteration.



Disclaimer: I am not an expert on any of these beam line modelling packages and can only use the models provided to me.

Madx Benchmarks

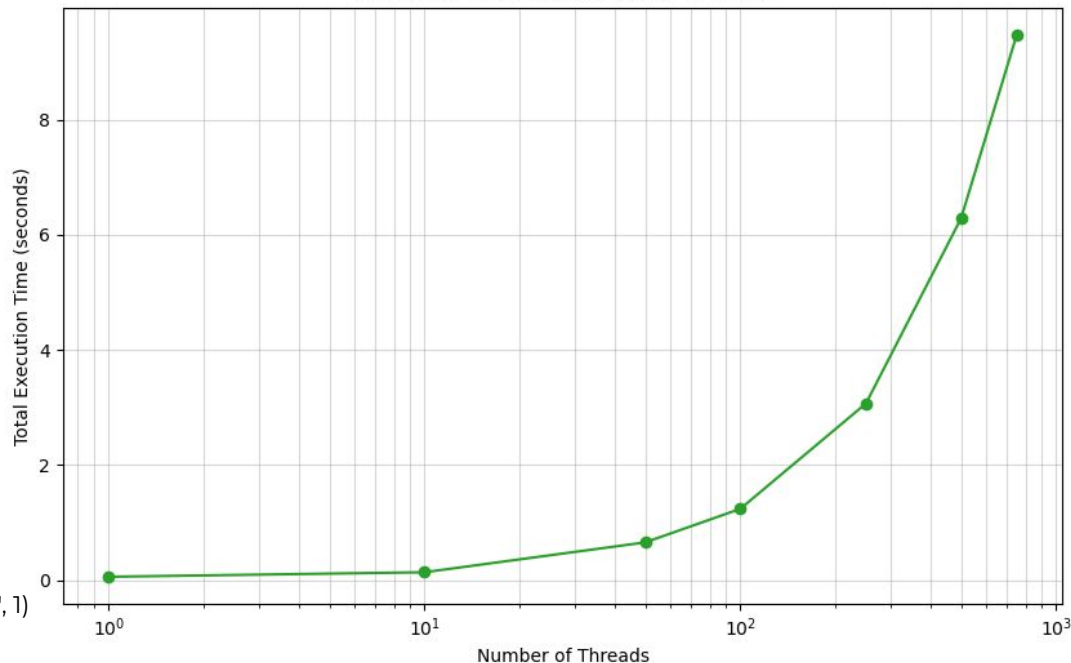
It was realized that my computer was 10 years old (Intel Xeon X5660). So we re-tested on the fastest computer we could find (12th Gen i7-12700 x 20).

Function Used in code:

```
from cpmad.madx import Madx
import random
def test_madx_function():
    madx = Madx(stdout=False)
    next_lattice_str = lattice_str.replace("4667.55224609375",
        str(random.randint(4000, 5000)) + ".55224609375", 1)
    madx.input(next_lattice_str)
    print("betx ->", madx.table.twiss.betx[0])
    madx.quit()
```

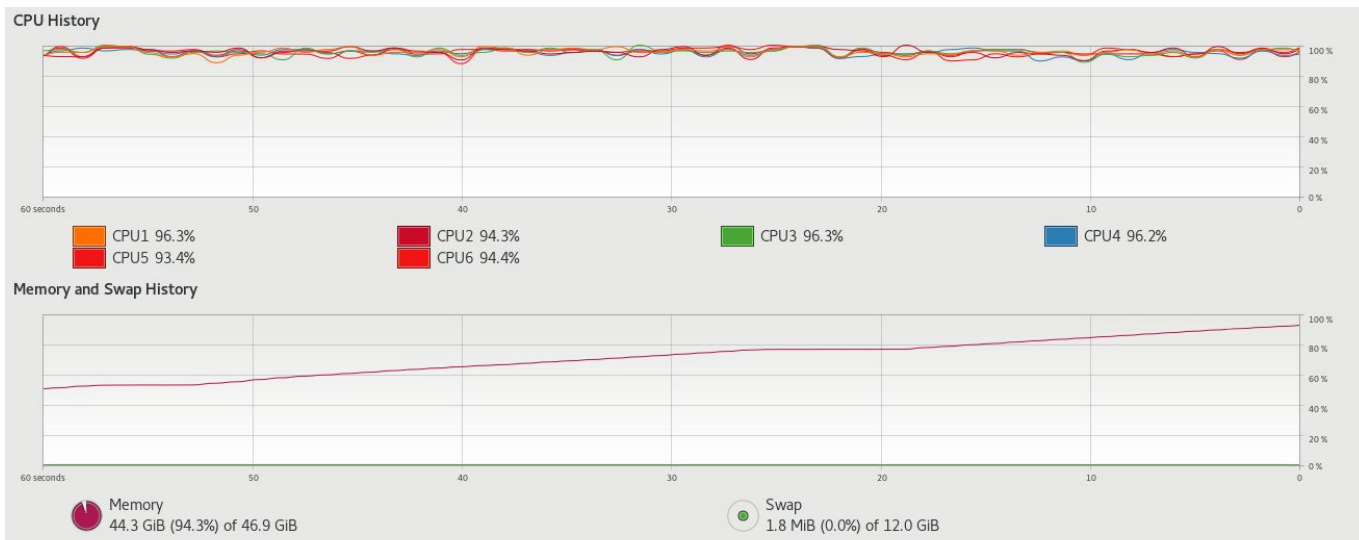
Source code: /home/cfsd/lbhajdu/SCR/bumpGUI/bumpGUI/benchmark_Madx.py

Execution Time vs. Number of Threads



Madx Benchmarks

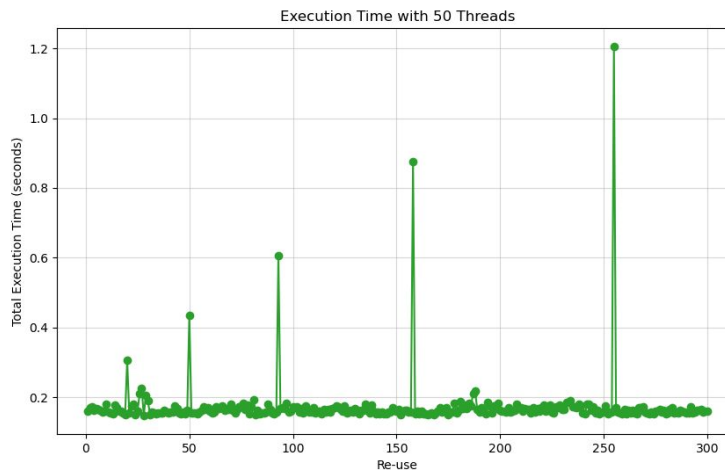
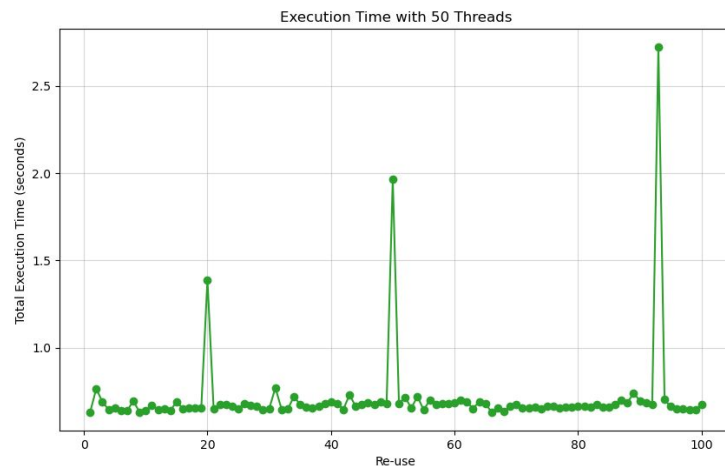
We tried to run in a re-entrant mode but it, or the cpymad.madx python package (more likely) leaked memory.



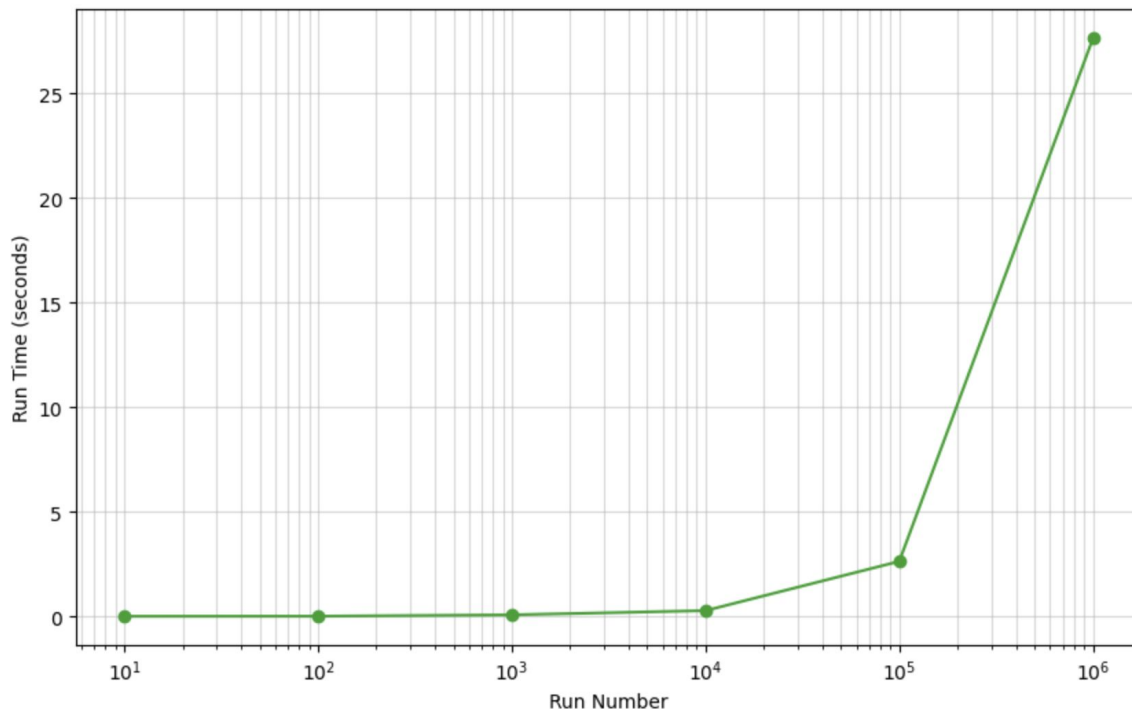
Madx Benchmarks

The unexplained

- One thread seems to run for much longer than the others, and it grows over time and is repeatable
 - Could it be swapping?
 - Process priority?
 - Garbage collection?
 - A bug?



SciBmad Benchmarks

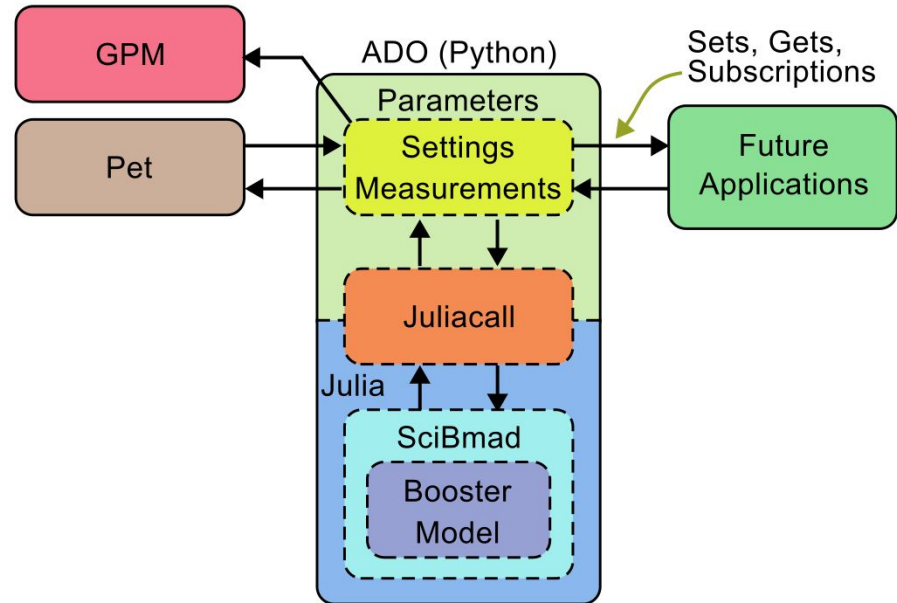


Test run by Jun Ma
I should re-run on same hardware

Current Proof of Principle

Reentrant SciBmad model from Lucy and Eiad, interfaceable via ADO (accelerator device object, CAD native control system)

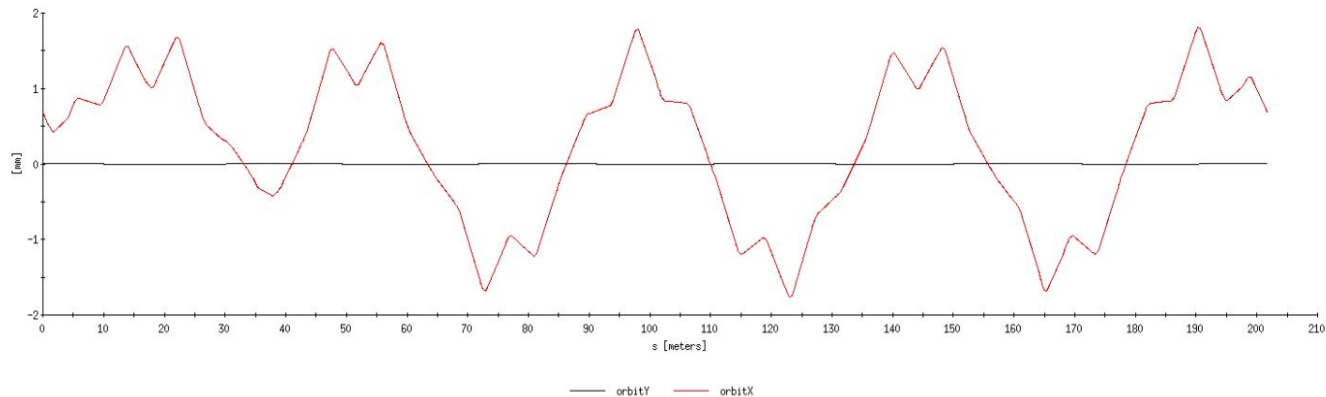
- Todo:
 - Expose more magnets, maybe other elements (RF) (currently hardcoded)
 - Parallel models for the ramp (next slides)
 - Update model (Lucy, Eiad)
 - What should trigger recalc? (Can't be one magnet change)
 - Do we use Lume? (upcoming slides)
 - Make Genesis deliverable (app)



File PPM Setup Logging Diagnostics

Fri Aug 28 13:18:24 2026, cycle 0

Booster Twin



Stop

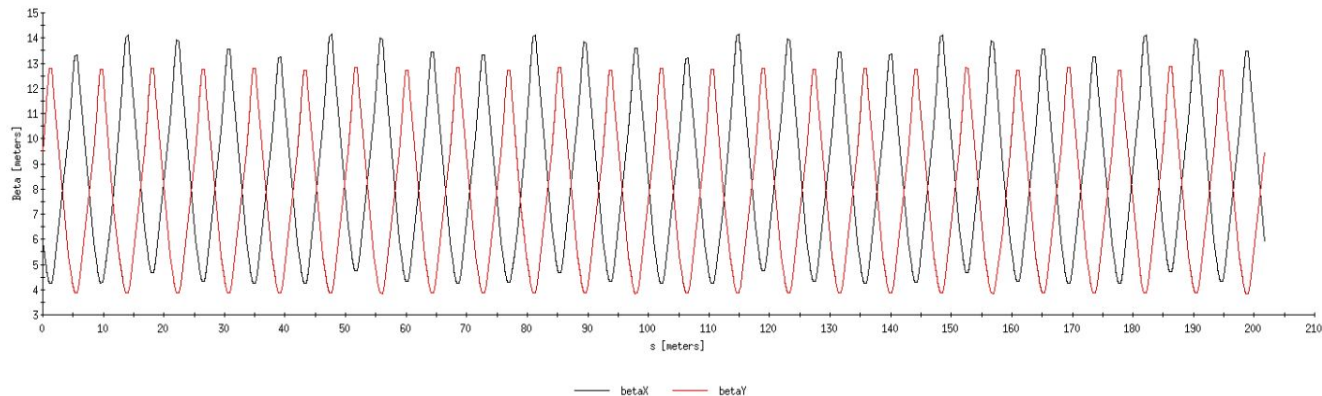
Jeff

Fregmeter

File PPM Setup Logging Diagnostics

Fri Aug 28 13:18:24 2026, cycle 0

Booster Twin



Stop

Page PPM Device Data Tools Buffer Help

system name booster.twin
generic name booster.twin
server name acnlinmf
ado class booster_twin

booster.twin	fecName	acnlinmf.pbn.bnl
booster.twin	description	Booster Digital
booster.twin	constructTime	1787256794
booster.twin	version	1.0 / ADO v5.0.0
booster.twin	className	booster_twin
booster.twin	modeS	Snap Ramp
booster.twin	timesS	[0 0 0 0 0 0 0 0]
booster.twin	sM	[0 0.276675...]
booster.twin	betaXM	[5.90834 5.3797...
booster.twin	betaYM	[9.48541 10.3414...
booster.twin	orbitXM	[0.000673107...]
booster.twin	orbitYM	[1.59897e-17...]
booster.twin	ic7S	12
booster.twin	id1S	20
booster.twin	id4S	10
booster.twin	id7S	5
booster.twin	ie1S	20

(18,3) booster.twin:ic7S Nudge: 1 466

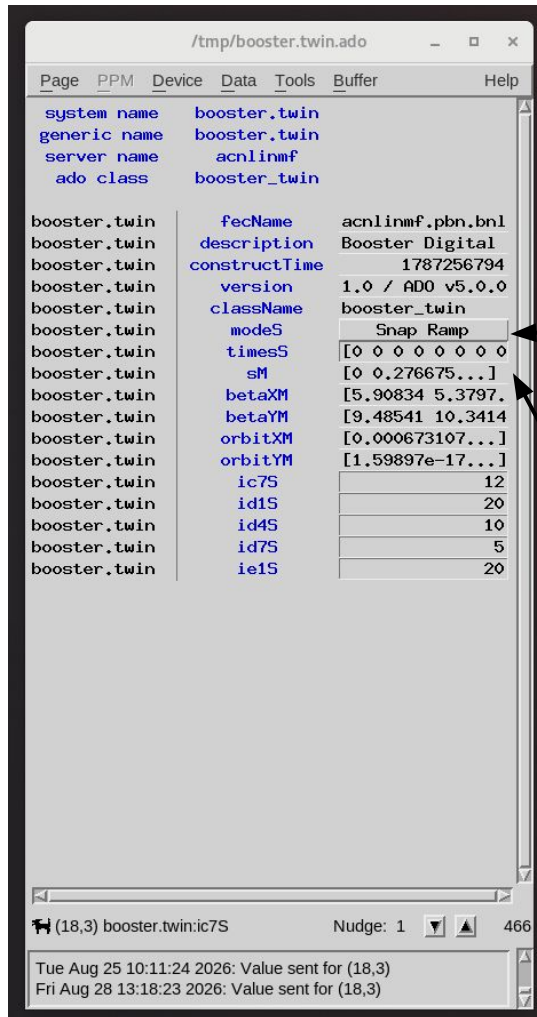
Tue Aug 25 10:11:24 2026: Value sent for (18,3)
Fri Aug 28 13:18:23 2026: Value sent for (18,3)

Features of Note

Modes of operation (not ready yet)

Modes:

- Live - the current state of the system
- Snap Ramp - CAD standard format file containing setpoints and readbacks
- Sandbox - User can adjust Time Slices (If we do it discretely)



The screenshot shows a software window titled "/tmp/booster.twin.ado" with a menu bar (Page, PPM, Device, Data, Tools, Buffer, Help). The main area displays system parameters for "booster.twin" in a table-like format. The parameters are organized into two columns. The first column lists parameters, and the second column lists their values. The parameters include system name, generic name, server name, ado class, fecName, description, constructTime, version, className, modeS, timesS, sM, betaXM, betaYM, orbitXM, orbitYM, ic7S, id1S, id4S, id7S, and ie1S. The values are displayed in a grid-like format. At the bottom, there is a status bar showing "(18,3) booster.twin:ic7S", "Nudge: 1", and "466". Below the status bar, there is a log area showing timestamps and values sent for (18,3).

Parameter	Value
system name	booster.twin
generic name	booster.twin
server name	acnlinmf
ado class	booster_twin
fecName	acnlinmf.pbn.bnl
description	Booster Digital
constructTime	1787256794
version	1.0 / ADO v5.0.0
className	booster_twin
modeS	Snap Ramp
timesS	[0 0 0 0 0 0 0 0]
sM	[0 0.276675...]
betaXM	[5.90834 5.3797...
betaYM	[9.48541 10.3414
orbitXM	[0.000673107...]
orbitYM	[1.59897e-17...]
ic7S	12
id1S	20
id4S	10
id7S	5
ie1S	20

(18,3) booster.twin:ic7S Nudge: 1 466

Tue Aug 25 10:11:24 2026: Value sent for (18,3)
Fri Aug 28 13:18:23 2026: Value sent for (18,3)

Triggering

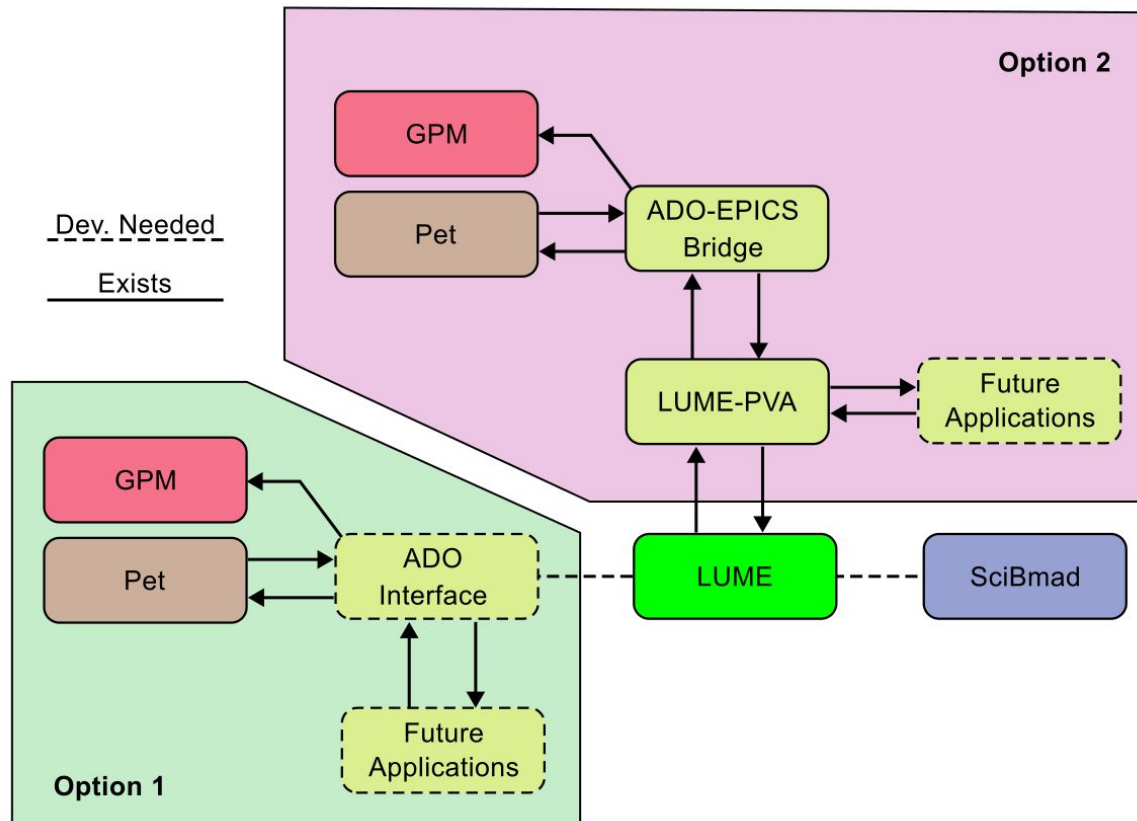
When should the model recalculate and broadcast new values?

- Any values change
 - A bad idea during a state restore (loading of archive)
- A specific Event-Link event (BT0)
- On request?

LUME

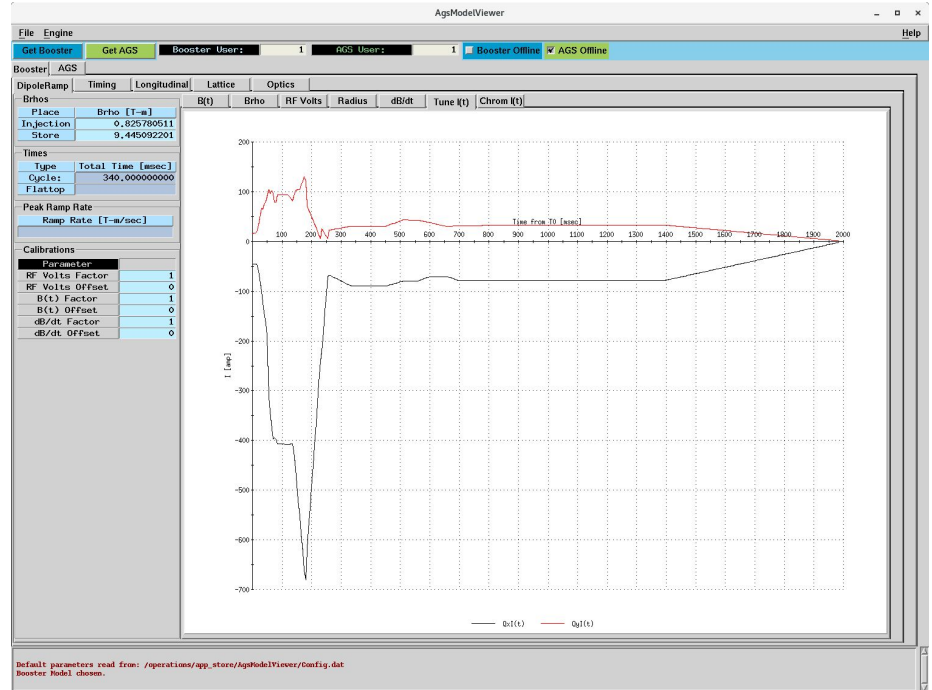
LUME is advantageous for letting us swap out the modelling software

- However, a SciBmad version does not exist
- On the control interface side, two options exist:
 - Implement an ADO interface similar to LUME-PVA (option 1)
 - Use existing LUME-PVA and ADO-EIPCS bridge(option 2)
- The December Genesis timeline is tight given all the deliverables needed; do we wait until phase 2 to integrate LUME?



High Level GUI Application

It has been suggested that we target AGS model viewer as something that can be replaced with a digital twin.



Question and Discussion

Fun Fact On Timing

Not all functions start at AT0, BT0 (booster time zero), there are hardware delay generators to broadcast events through the complex and most waveform generators are trigger by these events. But this is easy to look up in the control system (ADOs). EIC will be a bit more consolidated with software delay generators.

EventLinkDisplay: Booster Event Link

Setup Data EventMaps Diagnostics Help

Event Data For All PPM Users

LU1	LU2	LU3	LU4	LU5	LU6
LGB	LGE	LPP	BT0	BU1	BU2
BU3	BU4	BT1	BT2	BFR	BCE
BPP	AT0	AU1	AU2	AU3	AU4
BT3	BUE	APP	TT0	SCS	bmc.co_cor_st.rtg
bmc.co_cor_smpl.rtg	bmc.co_cor_sp.rtg	BPP_DELAYED	BRF_PSEUDO_PKR	bij.pseudo_pkr.rtg	bmc.co_cor_st.rtg
AU1new	AU2new	AU3new	AU4new	AU5	AU6
AU7	AU8	brf.harm_jmp2.rtg	brf.harm_jmp1.rtg	10HZ	TBEAM_ON
TBE	COG	NSRL	rtg_56_38	bgn.gnrc_real.rtg	bgn.spare_1.sp
bgt.jam_value.rtg	btg.tun_quad_st.rtg	btg.bgn_spill1.rtg	btg.end_spill1.rtg	NSRL_DELAYED	btg.prep_spill1.rtg
rtg_71_47	rtg_77_4d	60HZ	rtg_74_4a	rtg_75_4b	rtg_76_4c
rtg_77_4d	rtg_78_4e	bmd.tune_diag.rtg	bmd.kh_tun_st.rtg	bmd.kh_tun_sp.rtg	bmd.kh_tun_dschr.rtg
bmd.kv_tun_st.rtg	bmd.kv_tun_sp.rtg	bmd.kv_tun_dschr.rtg	bmd.tune_trig.rtg	bmd.kh_tun_smpl1.rtg	bmd.kv_tun_smpl1.rtg
bmc.sb_cor_st.rtg	bmc.sb_cor_smpl1.rtg	bmc.sb_cor_sp.rtg	BKH.TUNE_H	BKV.TUNE_V	B.F3_A5_TRIGGER
abi.pue_rd.brt	bmd.lpw_rdl1.rtg	btg.bcbw_cal.rtg	bmd.wdy_gt_str.rtg	bmd.wdy_gt_stp.rtg	bmd.gli_start.rtg
bmd.gli_start.rtg	bmv.late_read.rtg	bmd.pue_rd.rtg	bmd.freq_rd.rtg	rtg_162_a2	bmd.lpw_rdl.rtg
bmd.lossmon_rdl.rtg	bij.pseuok.gt	brf.rf_trak_st.gt	brf.early_off2.gt	brf.early_off2.gt	brf.synch_strt.gt
brf.xtrcn_strt.gt	bstr_norm_late.agt	bgn.spare_2.sp	bgn.late_cbw.sp	bgn.early_cbw.sp	bgn.integrate_p.sp
bmc.co_cor_st.rtg	bmc.co_cor_smpl1.gt	bmc.co_cor_sp.gt	bgt.upperlimit	brf.scope_trg1.gt	brf.scope_trg2.gt
brf.scope_trg3.gt	brf.scope_trg4.gt	abi.pue_rd.b.gt	bmd.kh_tun_st.gt	bmd.kh_tun_sp.gt	bmd.kh_tun_trig.gt
bmd.kv_tun_st.gt	bmd.kv_tun_sp.gt	bmd.kv_tun_trig.gt	bmd.tune_mtr_rd.gt	bmc.sb_cor_st.gt	bmc.sb_cor_smpl1.gt
bmc.sb_cor_sp.gt	bgt_93_5d	bgn.gs_trg_up_2.gt	BU1new	BU2new	BU3new

ACQUIRE DATA

Acquire Once

Acquire Continuous

Acquire From: / To:

SET/RESET

☐ Latch Display

Reset Display

Reset Counter

MONITOR LINK

- ✓ RHIC Event
- ✓ Blue Beam Sync
- ✓ Yellow Beam Sync
- ✓ RHIC Soft Event
- ✓ AGS Event
- ◆ **Booster Event**
 - ✓ Booster (Tandem PPM)
- ✓ Linac Event
- ✓ EBIS (EBIS PPM)
- ✓ EBIS (ETB PPM)
- ✓ CEC Beam Sync
- ✓ LERec Beam Sync

Event Name	Code(dec hex)	Event Time	(tmDiff)	uSecs-Sync	Count	PPM
10HZ	50	0x32	Thu Jul 13 13:51:29 2023	13,193us	1	
BU1new	161	0xA1	Thu Jul 13 13:51:29 2023	36,805us	49,998us	1 BU1new
BU1	11	0x0B	Thu Jul 13 13:51:29 2023	1us	50,000us	1 BU1
BPP	19	0x13	Thu Jul 13 13:51:29 2023	16,664us	66,664us	1 BU1
BPP DELAYED	35	0x23	Thu Jul 13 13:51:29 2023	15us	66,680us	1 BU1
BC1	238	0xEE	Thu Jul 13 13:51:29 2023	16,651us	83,331us	1 BU1
BFR	17	0x11	Thu Jul 13 13:51:29 2023	16,665us	99,997us	1 BU1
10HZ	50	0x32	Thu Jul 13 13:51:29 2023	13,192us	113,190us	2 BU1
BT0	10	0x0A	Thu Jul 13 13:51:29 2023	3,473us	116,663us	1 BU1
btg.tun_quad_st.rtg	65	0x41	Thu Jul 13 13:51:29 2023	1us	116,665us	1 BU1
bgt.jam_value.rtg	64	0x40	Thu Jul 13 13:51:29 2023	16us	116,681us	1 BU1
bmc.sb_cor_st.rtg	99	0x5A	Thu Jul 13 13:51:29 2023	1,982us	118,664us	1 BU1
bij.pseudo_pkr.rtg	37	0x25	Thu Jul 13 13:51:29 2023	7,499us	126,164us	1 BU1
bmc.co_cor_st.rtg	32	0x20	Thu Jul 13 13:51:29 2023	499us	126,664us	1 BU1
bmc.sb_cor_smpl1.rtg	91	0x5B	Thu Jul 13 13:51:29 2023	2,000us	128,664us	1 BU1
bgn.gs_trg_up_1.gt	223	0xDF	Thu Jul 13 13:51:29 2023	19,479us	148,143us	1 BU1
brf.scope_trg2.gt	144	0x99	Thu Jul 13 13:51:29 2023	8,455us	156,598us	1 BU1
bmd.wdy_gt_stp.rtg	102	0x66	Thu Jul 13 13:51:29 2023	65us	156,664us	1 BU1
bgn.integrate_p.sp	138	0x8A	Thu Jul 13 13:51:29 2023	388us	156,972us	1 BU1
bmd.tune_diag.rtg	89	0x59	Thu Jul 13 13:51:29 2023	9,593us	166,565us	1 BU1
bmd.lossmon_rdl.rtg	122	0x7A	Thu Jul 13 13:51:29 2023	20,582us	167,067us	1 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	1,597us	188,665us	1 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	188,715us	2 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	188,765us	3 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	188,815us	4 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	49us	188,865us	5 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	188,915us	6 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	188,965us	7 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	49us	189,015us	8 BU1
rtg_162_a2	108	0x6C	Thu Jul 13 13:51:29 2023	50us	189,065us	9 BU1

Booster Event Link has been selected