LUME

Lightsource Unified Modeling Environment

Christopher Mayes, Andrea Dotti, Chuck Yoon, Paul Fuoss.

October 21, 2018





Since first light at LCLS, there has been continuous invention of new operating modes, introduction of new optical elements, and rapid improvement in detectors. While these improvements have led to new experiments with much greater scientific impacts, their transfer to user operations has often taken several experimental runs (many months to years). The integration of these technical advances into scientific programs would be greatly accelerated by a modeling tool that allowed for quantitative assessment of the impact on scientific programs of facility improvements.

We propose develop the Lightsource Unified Modeling Environment (LUME) for unified modeling of X-ray free electron laser (XFEL) performance. This modeling tool will be built in several stages with an initial focus on quantitative prediction of critical parameters of the X-ray pulses delivered to experimental stations. This initial development will be followed by incorporation of X-ray–sample interaction and detector performance. This project will take a holistic approach starting with the simulation of the electron beams, to the production of the photon pulses and their transport through the optical components of the beamline, their interaction with the samples and the simulation of the detectors, followed by the analysis of simulated data.

LUME will leverage existing, well-established codes [Astra, Bmad, Elegant, Genesis, Impact for electrons, Genesis 1.3 for FEL simulation, and the "Synchrotron Radiation Workshop" (SRW) for X-ray optics] that will be driven and configured by a coherent high-level framework. The high-level framework will build on the Simex platform being developed by the European Cluster of Advanced Laser Light Sources (EUCALL). The platform will be built with an open, well-documented architecture so that science groups around the world can contribute specific experimental designs and software modules, advancing both their scientific interests and a broader knowledge of the opportunities provided by the exceptional capabilities of X-ray FELs

LUME will be the first platform in the world for unified modeling of XFEL performance. LUME's optimization capabilities will guide SLAC accelerator physicists in developing world leading XFEL performance. LUME will identify performance bottlenecks, both in the accelerator and photon transport, and enhance operational efficiency and reliability. The complete integration of electron and X-ray processes will allow LCLS scientists to invent instruments that optimally use those unique X-ray beams. Finally and most importantly, the ability to simulate experiments will stimulate the development of new approaches to the scientific and technological challenges facing the country, maximizing the impact of DoE's investment in cutting-edge X-ray sources.

Goal for Start-to-End Simulations

- How should the accelerator be configured to produce the "best" pulses?
- How do real pulses, instead of idealized pulses, propagating through the instrument interact with the optics and the sample?
- What FEL components and configurations degrade, or enhance, the experimental results?

The Lightsource Unified Modeling Environment (LUME) will provide answers to important questions and solutions to design problems. Average over 200 shots









- Since first light at LCLS, there has been continuous invention of new operating modes, introduction of new optical elements, and rapid improvement in detectors.
- While these improvements have led to new experiments with much greater scientific impacts, their transfer to user operations has often taken several experimental runs (many months to years).
- Within a single experimental shift (12 h), the majority of time can be spent on setup and tuning (even for an established operating mode).
- The integration of these technical advances into scientific programs would be greatly accelerated by a modeling tool that allowed for quantitative assessment of the impact on scientific programs of facility improvements.

Approach: Lightsource Unified Modeling Environment (LUME)



- LUME will be a simulation platform that glues together simulation modules in an integrated pipeline.
- Wrap standard, developed electron/photon simulation codes with a common interface (Python)
- Use the newly developed openPMD standard for data exchange (HDF5 files)
- Well documented for use by developers, <u>Ph.D. students</u>, users (simulation as a service)
- Supporting technologies: Machine database, Simulation Catalogue, Visualization and (G)UI,Optimization hooks, PSANA integration
- Expands on the developed **SimEx** (Eu. XFEL) platform (C. Yoon, C. Fortmann-Grote, ...)
- Will prototype with immediate use cases

SLAC

An efficient, high-fidelity and simple-to-use simulation platform will allow:

- Prototyping of novel accelerator operation modes without using valuable machine time.
- Accelerator and FEL parameters to be optimized with the guidance of simulations, consequently improving the efficiency of data-collection
- Determination of corrections and calibrations needed to accurately interpret experimental data.
- Early development of analysis algorithms to:
 - enhance quasi-real time data visualization.
 - improve the quality of the data collected.
 - reduce the time needed to obtain scientific results.
- Prioritization of facility development by identifying parameters that critically affect scientific success.
- Development of stronger science case by testing experimental assumptions.

Goal is to guide experiment design and interpretation of results, not to predict the results of specific experiments.

Competition/Alternatives

SLAC

LUME is a medium-scale project (4-8 FTEs, 3-5 years). Nothing of its kind currently exists. Alternatives are:

- Hand-stitch (script) start-to-end simulations as needed
 - Requires expert(s) for each step
- Model-free tuning (direct optimization) of the live machine
 - + Time saved on modeling
 - Limited to local optimization
 - Not necessarily fast
- Surrogate modeling (neural network) of the as-built machine
 - + Fast switching between established operating modes
 - + Time saved on modeling
 - Limited by data acquired from a limited set of operating conditions

Analysis Workflow

- 1. Simulation of FEL Source
- 2. Transport of X-rays to Experiment
- 3. Focusing X-Rays (e.g. KB mirrors or Be Lens)
- 4. Measurement of Wavefront using the Talbot effect

Work done by: Yanwei Liu, Matthew Seaberg, Yuantao Ding, Gabriel Marcus, Daniele Cocco, Anne Sakdinawat

Use Case: Testing Single Shot Wavefront Sensor (1/4)



Genesis Simulation



Genesis Output to Wave Propagation

SLAC



Use Case: Testing Single Shot Wavefront Sensor (2/4)







Simulated Distorted Wavefront from HOMS Mirror

SLAC



Simulation parameters: phase grating pitch = 20 um, photon energy = 9.5 keV, working distance = 1160 mm (n=3), Detector resolution = 2 um, detector pixel equivalent size = 0.6 um.

Use Case: Testing Single Shot Wavefront Sensor (3/4)



SLAC



To accommodate the divergent focused beam, the phase grating has a pitch of 14um, placed 0.400 meters after the focus. Detection was at the 1st Talbot plane at 0.354 m away from the grating. We assumed an image blur of 2um and pixel size of 0.6um. Photon energy is 9.5 keV.

Use Case: Testing Single Shot Wavefront Sensor (4/4)







Image Pixelization

SLAC



LUME Pipeline



- 1) Simulation of electron-beam production.
- 2) Modeling of the accelerator with known component performance and errors.
- Simulation of X-ray production in undulator including seeding.
- 4) Propagation of X-rays through the beamline optics.

- 5) Calculation of the photon-matter interaction at the sample.
- 6) Selection and propagation of photons and particles to the detector.
- 7) Modeling of the detector response.
- Test reconstruction and data analysis algorithms with simulated detector signals.

Initial LUME Module Codes





SLAC

LUME will be a *simulation platform* that glues together *simulation modules* in an integrated pipeline. This platform will:

- Track the creation of the electron bunch at the cathode and follow it through the accelerator.
- Create an X-ray pulse from bunched electrons via the FEL interaction in the undulator (including seeding).
- Propagate the X-ray wavefront through the X-ray beamline.
- Perform atomistic modeling of the photon-material interaction.
- Track the resultant diffracted and emitted particles (photons, electrons and ions) to appropriate detectors.
- Assemble the simulation results into three dimensional datasets that can be used for algorithm development.

The simulation modules will be reused from existing efforts. New modules will only be developed to fill clear needs where solid benefit can be demonstrated.

A. Simulation framework

The LUME backbone will be a framework that leverages already well established calculation codes interfaced to each other via standard data formats and APIs. To maximize efficient use of existing resources, we plan to use of existing software solutions (e.g. SimEx from the EUCALL consortium), proactively participating in their development, proposing features and supporting the parts of the code of main interest to SLAC.

- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation



SLAC

A. Simulation framework

B. Machine Description Database

In order to efficiently simulate the performance of a given beamline, a database of electron and x-ray optics and associated hardware will be created. In addition to supplying information for simulations, this database should find general applicability for design, installation and maintenance work.

- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation

- A. Simulation framework
- B. Machine Description Database

C. Simulation Data Catalog

The data catalog provides a repository and for metadata cataloguing of simulation results. The goal is to increase simulation efficiency by capability to reuse, in part or fully, simulation data. The data catalog is independent of the simulation framework but its main client will be simulation workload, and as such it needs to be designed closely with Task A.

- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation



- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog

D. Visualization and (G)UI

The framework provides a visualization subsystem defining what can be visualized. Visualization is implemented in separate modules via visualization drivers. We prefer web-based approaches that do not require additional software and graphic libraries.

- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation



Iamp and dream simulation

OASYS - OrAnge SYnchrotron Suite



- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI



E. Numerical Optimization

The goal is to maximize experimental performance, not to deliver beam with intermediate qualities at artificial code transition points. Experiments can subtly depend on properties developed far upstream that are difficult to quantify. With LUME, these artificial interfaces can be removed and the composite system can be optimized as a whole.

- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation

- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization

F. Particle File Format



Use openPMD to describe Particle and Mesh data. Researchers from HZDR, LBNL, DESY, Cornell University, and SLAC are defining an Accelerator and X-ray extension. The LUME task is to develop interface code for simulation tools (e.g. the Astra space charge code), so that the framework can use the format to glue together different codes.

- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation

- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format

G. Integration with PSANA

We want to be able to produce data-formats in LUME that are the same as those produced by the LCLS experiments. The challenge is that we cannot put these functionalities inside the "Core" of the simulation platform if we want other facilities to share the same "Core" software.

- H. Immediate Use Cases
- I. Documentation





- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases

LUME will use examine high-impact problems currently facing development projects at LCLS-II, in particular, LCLS and SLAC to drive the development of the modeling capabilities.

500 eV Oasys Tools DREAM beamline Shadow ALS Sources filor motion effect Shadow Sources eometrical Sour Shadow ALS Optical Elements 1200 eV Ellipsoid Micro Ellipsoid I Plane Spherical Toroidal Mirror Mirror eometrical Source ... General Options ots Output ✓ Automatic Execution Plotting Style Select level of Plotting Detailed Plot Run Shadow/Tra **Report Eight** KZ X',Z' X,X' Z,Z' Energy Footprint Position Basic Setting Advanced Setting 🗙 🛷 🗢 🛝 🚱 🥘 🗎 👳 Reflectivity Dimensions 17406.0 Surface Shape Paramete 🔭 — | 🔪 🗰 W: 1 🔅 Total Ray 25000 Type internal/calculated 17406 250 Focii and Continuation Plane Different Total Lost Rave Object Side Focal Distance [mm] 5250.0 0.1 EWHM X [um] 0.2344 \bigcirc Image Side Focal Distance [mm] 800.0 0 1036 Incidence Angle -0.1 -0.3 Shadow Special Elements Cylinder Paramete 30805050000 -6.4 -6.2 -6.0 -5.8 -5.6 -5.4 Shadow PostProcesso Surface Curvature Concave * X [um] Frequency Cylindrical 🔍 # T 🖊 II Cylinder Orientation (deg) [CCW from X axis] 0 -800 600 Calculated Surface Shape 400 3025.0 Calculated semi-major Axis [mm] Calculated semi-minor Axis (mm) 43.0265 Render Surface Shape

...

I. Documentation

- A. Simulation framework
- B. Machine Description Database
- C. Simulation Data Catalog
- D. Visualization and (G)UI
- E. Numerical Optimization
- F. Particle File Format
- G. Integration with PSANA
- H. Immediate Use Cases
- I. Documentation

LUME must have high quality, complete documentation to be generally usable by our target audience.

Design Philosophy: Overall

LUME simulation for a specific facility (user code): configurations, specification of applications, filenames, analysis routines



External Dependencies

Who are the USERS?

- 1) FEL Developer
- 2) PhD Graduate Student
- 3) FEL User (Simulation as a service)

Design Philosophy: Code Base

- Pure Python wrappers
 - Platform independence
 - Minimal complexity
- Dependencies are scoped
 - Can simulate FEL without X-ray optics
 - Can simulate X-ray optics from canned FEL output
 - Interactive configuration manager (e.g. Conda)
- Work with underlying code developers to achieve better portability
- Container distribution (e.g. Docker or Singularity) to facilitate cloud and highperformance computing



- markup / schema for hierarchical data files
- truly, scientifically self-describing
- basis for **open data workflows**
- high-level description
- human and machine readable
- particle and mesh based data
- general unit system
- file format agnostic
- portable
- scalable from desktop to HPC
- forward-updatable
- https://github.com/openPMD

HDFView 2.9	
<u>File W</u> indow <u>T</u> ools <u>H</u> elp	
差 🗂 🔌 🖪 🗂	
Recent Files /home/axel/progg	en_partiell/openPM
s example.h5	
🛉 📹 data	
🔶 📹 0	
🛉 📆 fields	
🔶 🛀 E	
— 鼲 rho	
🕈 📹 В	
— 🏙 ×	
— 🏙 у	
— 🎘 z	
🕈 📹 particles	
electrons	
— 🔙 charge	
🗠 🛀 mass	
🗠 🞑 momentum	
r 🖼 position	
— 🌐 ×	
- HE Y	
⊂ 8⊞ Z	
— 🧱 weighting	
electrons (63328, 4)	
Group size = 5	
currentDeposition = Esirk	epov
longName = My first electr particleInterpolation = Tril	on species inear
particlePush = Boris	
particleShape = 3.0 particleSmoothing = none	
,	
Log Info Metadata	

openPMD standard (1.0.0, 1.0.1, 1.1.0) the underlying file markup and definition A. Huebl et al., doi: https://doi.org/10.5281/zenodo.1345077 PIConGPU (HZDR) electro-dynamic particle-in-cell code maintainers: A. Huebl, M. Bussmann et al. Warp (LBNL, LLNL) electro-dynamic/static particle-in-cell code maintainers: J-L. Vay, D. Grote, R. Lehe et al. FBPIC (LBNL, DESY) spectral, fourier-bessel particle-in-cell code maintainers: R. Lehe, M. Kirchen et al. SIMEX Platform (EUCALL, European XFEL) simulation of advanced photon experiments maintainer: C. Fortmann-Grote Smilei (Maison de la Simulation) electro-dynamic PIC maintainers: M. Grech et al. UPIC-Emma 2.0 (UCLA) spectral PIC maintainers: F. Tsung et al. **Osiris** (IST, UCLA) electro-dvnamic PIC maintainers: F. Tsung, R. Fonseca et al. **Bmad** (Cornell): *particle accelerator* maintainers: D. Sagan et al.

SLAC

Framework development: Genesis 1.3 v2

- Python wrappers have been developed for the Genesis 1.3 FEL code.
- Code repository: <u>https://github.com/slaclab/lume-genesis/</u>
- Examples in Jupyter notebooks.
- Code works on Cori at NERSC entirely through the web browser.
- Code is now incorporated into the workflow of several researchers.
- Interactive visualization using the bokeh plotting library (originally developed for data science).
- Wrappers are also being developed for:
 - Impact-T (injector code)
 - Bmad/Tao (accelerator code)
 - SRW/WPG (X-ray code)

	ו	Chttps://jupyter-dev.nersc.gov/user/cmayes/ C
File Edit View Run Kerne	Hub Tal	os Settings Help
+ 🗈 ± C	E Launc	her × Rexample interactiv Bokeh interactive ×
♠ > > cori > genesis	8 + %	C D + C Markdown - Python [default]
Name t Loot		
Mame Last Modified	[2]:	<pre>from genesis import genesis, parsers, interactive from bokeh.plotting import show, output_notebook, output_file</pre>
 bin 11 days ago examples a month ago 		<pre>import os output_notebook()</pre>
🗅 Genesis_Cur 11 days ago		BokehJS 0.13.0 successfully loaded.
test 2 hours ago		•
work 2 hours ago		
B edison instra a month ago	[3]:	# Point to my genesis binary, working directory
D genesis sou a month ago		<pre>params = { 'workq1r':os.path.expanovars('\$HUME/corl/genesis/Work('), 'genesis bin':os.path.expanovars('\$HUME/corl/genesis/bin/genesis2').</pre>
genesis_sou a month ago		<pre>general_out rospectrespendents(state correspondents) out/generals(); }</pre>
		g = genesis.Genesis(**params)
	L	
	[4]:	# Turn on particle output
		# Turn on field output
		<pre>g.input_params['idmpfld'] = 1 # .dfl file</pre>
		g.input_params['npart'] = 2048
		# Turn on history
		iskip = 10
		<pre>g.input_params['ippart'] = iskip # .par file #Write the particle distribution to file at each</pre>
		g.input_params['ipradi'] = iskip # .fld file #Write the field distribution to file at each IP
		dgrid = g.input_params['dgrid']
	[5]:	g.run_genesis()
		<pre>init dir: /global/u2/c/cmayes/cori/genesis/test running genesis in /global/u2/c/cmayes/cori/genesis/work/genesis_run_LqFDD284s6 <class 'genesis.genesis.genesis'="">.write_lattice = WARNING: ran out of quads for lattice</class></pre>
		Genesis 1.3 has begun execution
		Diasca antar input file name
		*** Warning: external magnet definition too short for quadrupole field
		*** Warning: Size of particle file [Mbytes]: 14
		wew Warning: Size of radiation file [Mbytes]: 150
		Slice 1: Simulation 10% completed.
		Slice 1: Simulation 30% completed.
	1	Slice 1: Simulation 40% completed.
		Slice 1: Simulation 50% completed.
		Slice 1: Simulation 70% completed.
	1	Slice 1: Simulation 80% completed.
		Slice 1: Simulation 90% completed.
		*** Writing history record for slice 1
		Genesis run has finished

Framework development: Bmad/Tao

- Bmad is an open-source software library for charged particle simulations.
- www.classe.cornell.edu/bmad/
- Currently developing direct openPMD compatibility through BeamPhysics extension
- Tao us built on Bmad as a interactive design/simulation tool with command-line input and a plot window.
- Tao can now be called from Python, with direct interaction.
- Special Tao commands to acquire formatted data, form plots in Python.
- Works locally and remotely.



Tao command window

	import os	1						
	Inititialize Tao							
2]:	<pre># Point to local installat BASE_DIR=os.environ['ACC_R print('Bmad installation:</pre>	ion DOT_DIR'] ', BASE_DIR)						
	<pre># Pick an example init root = BASE_DIR+'tao/examp os.chdir(root) init = root+'tao.init' # Make tao instance tao=Tao(init)</pre>	les/cbeta_ffag/'						
	Bmad installation: /Users	/chrisonian/Code/b	mad_svn/					
	Sanda common	d						
	Send a comman	a						
]:	# Tao command	informed tool).						
	Jupyter magic %	‰tao						
	Jupyter magic %	6 %tao nd commands to Tao	directly in t	he jupyter	noteboo	k, using	the %%t	tao magic.
5]:	Jupyter magic % This is an alternative way to see Mtao sho lat 1:10	5%tao nd commands to Tao	directly in t	he jupyter	noteboo	k, using	the %%t	tao magic.
5]:	Jupyter magic % This is an alternative way to see Mtao sho lat 1:10	5 %tao nd commands to Tao	directly in t	he jupyter	noteboo	k, using	the %%t	ao magic.
5]:	Jupyter magic % This is an elternative way to se Witao sho lat 1:10 Tao> sho lat 1:18 ¥ Values shown are for the	5%tao nd commands to Tao Exit End of each	directly in the second se	he jupyter	noteboo	k, using	the %%t	ao magic.
5] :	Jupyter magic % This is an alternative way to see %tao sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the #Index name k	by tao and commands to Tao Exit End of each Ey	directly in the din the directly in the directly in the directly in the direct	he jupyter	noteboo beta a	k, using phi a	the %%t eta a	orbit x [mm]
5] z	Jupyter magic % This is an alternative way to se Notao sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the # Index name 1 MAR.BEG	5%tao hd commands to Tao Exit End of each ey arker	directly in the dintedirectly in the directly in the directly in the directly	he jupyter ۱ ۹.000	noteboo beta a 0.73	phi 0.000	eta a -0.02	orbit x [mm] -12.850
5] :	Jupyter magic % This is an alternative way to se Witao sho lat 1:10 Tao- sho lat 1:10 ¥ Values shown are for the #Index name 1 MAR.BEG M 3 FA.MAR.BEG M 3 FA.MAR.BEG M	5%tao hd commands to Tao Exit End of each ey arker arker arker	directly in the second	he jupyter l 0.000 0.000	noteboo beta a 0.73 0.73 0.73	phi a 0.000 0.000	eta a -0.02 -0.02	orbit x [mm] -12.850 -12.850
5]:	Jupyter magic % This is an alternative way to see %tao sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the # Ualues shown are for the # Index name # AnAMERG M 3 FA.ALIGN_PATCH P 4 FA.PIPPIA P	book tao and commands to Tao Exit End of each ey arker arker arker tch be	directly in the second	he jupyter l 0.000 0.000 0.050	noteboo beta a 0.73 0.73 1.38	phi a 0.000 0.000 0.000 0.000	eta a -0.02 -0.02 -0.02 -0.02	orbit × [mm] -12.850 -12.850 -17.025
5] =	Jupyter magic % This is an alternative way to se VAtao sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the # Undex name f I MAR.BEG M 2 FA.MAR.BEG M 3 FA.ALION.PATCH P 4 FA.PIPPBIA P 5 FA.QUABIEL 0	6%tao the commands to Tao Exit End of each av av tch tch tpp tadrupple	directly in ti Element: 5 0.000 0.000 0.000 0.000 0.000 0.100	he jupyter l 0.000 0.000 0.000 0.055	noteboo beta a 0.73 0.73 0.73 1.38 1.87	phi a 0.000 0.000 0.119 0.198	eta a -0.02 -0.02 -0.02 -0.02 -0.02 -0.02	orbit × [mm] -12.850 -12.850 -17.025 -19.187
5] z	Jupyter magic % This is an alternative way to see %tao sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the # Values shown are for the # Index name # AnARAR.EM # FA.AHAR.EM # FA.AHAR.EM # FA.AHAR.EM 6 FA.QUABJER.ID 6 FA.QUABJERA.HD	6%tao and commands to Tao exit End of each exit End of each exit pricer tach tippe prioriter prioriter adrupple arker	directly in th Element: 5 0.000 0.000 0.050 0.055 0.108	he jupyter l 0.000 0.000 0.050 0.050 0.050	noteboo beta a 0.73 0.73 1.38 1.87 1.87	phi a 0.000 0.000 0.119 0.198 0.198	eta a -0.02 -0.02 -0.02 -0.02 -0.01 -0.01	orbit x [mm] -12.850 -12.850 -12.850 -17.025 -19.187
5]:	Jupyter magic % This is an alternative way to see Notao Sho lat 1:10 Taos sho lat 1:10 Taos sho lat 1:10 # Values shown are for the # I MAR.BEG M 2 FA.MAR.BEG M 3 FA.ALION_PATCH P 4 FA.PIPBIA P 5 FA.QUABIAL	6%tao Ind commands to Tao Exit End of each ey arker a	directly in the second	he jupyter l 0.000 0.000 0.050 0.058 0.058 0.058	noteboo beta a 0.73 0.73 1.38 1.87 1.20	phi a 0.000 0.000 0.119 0.198 0.282	eta a -0.02 -0.02 -0.02 -0.01 -0.01 -0.01	orbit × [mm] -12.850 -12.850 -12.850 -12.850 -12.850 -12.850 -19.187 -19.187 -19.187
5] :	Jupyter magic % This is an alternative way to se Witao Sho lat 1:10 Tao- sho lat 1:10 Tao- sho lat 1:10 # Values shown are for the #Index name FAR, MRR, BEG FAR, MRR, BEG FAR, MRR, BEG FAR, MRR, BEG FAR, MRR, BEG FAR, MAR, MAR, BEG	S%tao and commands to Tao Exit End of each sy arker arker arker arker arker arker arker bitch lipe addrupple prevent	directly in the second	he jupyter l 0.000 0.000 0.050 0.058 0.058 0.058 0.058 0.058 0.058 0.025 0.025	noteboo beta a 0.73 0.73 1.38 1.87 1.20 0.80 0.80	phi a 0.000 0.000 0.119 0.198 0.198 0.282 0.344 0.344	eta a -0.02 -0.02 -0.02 -0.02 -0.01 -0.01 -0.01 -0.01 -0.03 -0.03	orbit x [mm] -12.850 -12.850 -17.0255 -19.187 -19.187 -14.861 -11.883 -11.883
5]:	Jupyter magic % This is an alternative way to see Shot lat 1:10 Tao> sho lat 1:10 # Values shown are for the # findex name for the findex name for the # Findex name for the findex name for the # Findex name for the findex name for the findex name for the # Findex name for the findex na	6%tao and commands to Tao Exit End of each by arker arker tarker	directly in ti	he jupyter l 0.000 0.000 0.050 0.055 0.025 0.025 0.025	noteboo beta a 0.73 0.73 1.38 1.87 1.87 1.87 0.80 0.80 0.50	phi a 0.000 0.000 0.119 0.198 0.282 0.344 0.344 0.434	eta a -0.02 -0.02 -0.02 -0.02 -0.02 -0.01 -0.01 -0.01 -0.01 -0.03 -0.03 -0.03	ao magic. x [mm] -12.850 -12.850 -17.850 -17.850 -17.025 -19.187 -19.187 -14.861 -11.883 -11.883 -8.904
5] =	Jupyter magic % This is an alternative way to see Vatao sho lat 1:10 Tao- sho lat 1:10 Tao- sho lat 1:10 # Values shown are for the # Unlex shown are for the # Unlex name # 2 FA.MAR.BEG M 3 FA.ALION.PATCH 9 FA.040312 0 6 FA.040312 P 9 FA.BM01 M 1 FA.PIP0182 P # Thefa.BM01 M	6%tao ad commands to Tao Exit End of each exit End of each by arker arker arker arker arker arker arker arker bige big	directly in ti	he jupyter l 0.000 0.000 0.050 0.055 0.058 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.025 0.000 0.005	noteboo beta a 0.73 0.73 1.38 1.87 1.87 1.87 0.80 0.80 0.80 0.50 beta	phi a 0.000 0.000 0.119 0.198 0.198 0.282 0.344 0.344 0.344 0.344	eta a -0.02 -0.02 -0.02 -0.01 -0.01 -0.01 -0.03 -0.03 -0.03 eta	orbit x [mm] -12.850 -12.850 -12.850 -12.850 -12.850 -12.850 -13.187 -14.861 -11.883 -11.883 -8.904 orbit
5] =	Jupyter magic % This is an alternative way to see State Sho lat 1:10 Tao> sho lat 1:10 # Values shown are for the # Undex name findex name findex name findex name findex name findex name findex name findex name	6%tao Ind commands to Tao and commands to Tao Exit End of each arker	directly in ti Element: s 0.000 0.000 0.050 0.108 0.108 0.108 0.190 0.190 0.191 0.191 5 s	he jupyter l 0.000 0.050 0.058 0.008 0.058 0.008 0.058 0.025 0.025 1	noteboo beta a 0.73 0.73 1.38 1.87 1.20 0.80 0.50 beta a	phi a 0.000 0.000 0.119 0.198 0.282 0.344 0.344 0.344 0.344 0.344 0.344	eta a -0.02 -0.02 -0.02 -0.02 -0.02 -0.01 -0.01 -0.01 -0.03 -0.03 -0.03 eta a	orbi x [mm -12.85 -12.85 -12.85 -12.85 -12.85 -12.85 -12.85 -14.86 -11.88 -11.88 -11.88 -11.88 -8.90 orbi x [mm

Python wrapper for Tao

Interactive development in Jupyter

- Output parsed into Python enables rapid analysis and visualization
- Sliders work locally and remotely





Photon field phase evolution

Power profile evolution

Validation: LCLS Accelerator

- Setting quadrupole focusing (k1) according to design does not give good actual FEL performance.
- Quads are tuned empirically to get good performance.
- The accelerator model then predicts that the beam would be wildly defocused.
- It is not understood why these settings work.
- The model needs to be calibrated against measurements.



Validation: Development of Single Particle Imaging

A key step in establishing a trusted simulation platform is validating the simulated data with real experimental data from a beamline and understanding how large the errors can be.

Example of simulating a Single Particle Imaging (SPI) experiment at LCLS. Preliminary results from Haoyuan Li, Philip Hart, and Chuck Yoon:

- RDV data was collected as part of the SPI initiative in Aug 2015 at AMO beamline on pnCCD cameras.
- Simulated diffraction pattern of an RDV from the AMO beamline setup using pysingfel package. The simulation
 assumes 10¹² photons per pulse in a 1.5 micron focus. Detector noise was added using dark measurements from
 the experiment. The total number of photons is in the correct ball park (~400K photons per image).



32

LUME - Lightsource Unified Modeling Environment



Since 1st light at LCLS, there has been continuous:

- Invention of new operating modes
- Introduction of new optical elements
- Rapid improvement in detectors

A start-to-end model will increase LCLS impact by:

- Guiding physicists in developing world leading XFELs
- Identifying performance bottlenecks
- Enhancing operational efficiency and reliability

Unified modeling of electron and x-ray processes will:

- Enable the invention of instruments to fully exploit XFELs
- Stimulate new experimental approaches and maximize the impact of DoE investment.

LUME Features

Leverages on existing capabilities, for example:

- ASTRA, Bmad, Elegant, and Impact codes for charged particles.
- Genesis 1.3 code for the FEL interaction
- Synchrotron Radiation Workshop (SRW) (Brookhaven)
- SimEx (simulation platform started by EuCALL)
- openPMD particle mesh data files (HZDR, LBNL, et al.)

Open source, modular, well documented design

- Efficient development with emphasis on code reuse
- External science groups can contribute
- Capabilities evolve with XFEL capabilities and science



END