

DANSE: Distributed Data Analysis for Neutron Scattering Experiments

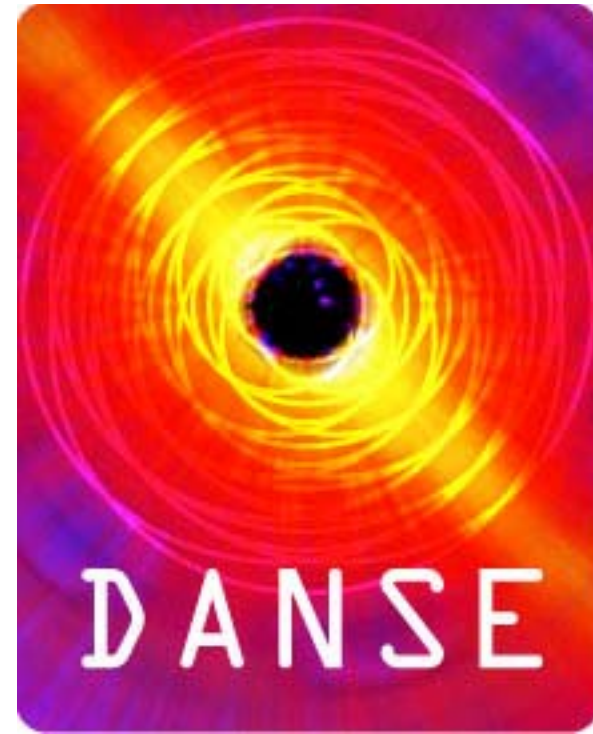
Project Overview

Brent Fultz

California Institute of Technology

- Goals
- Operations, Organization, Expectations
- Progress and Plans

A Decision Was Made by
Project-Wide Vote



Close Election!

DANSE



DANSE Software will enable:

New Science

Better Science

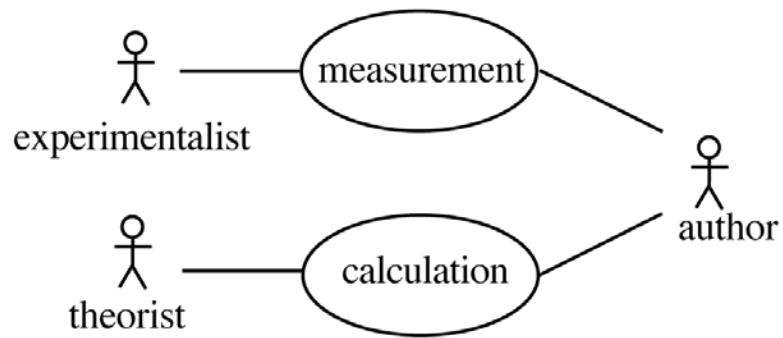
Ease of Use

Software Stability and Reuse

Support Early Operations of the SNS

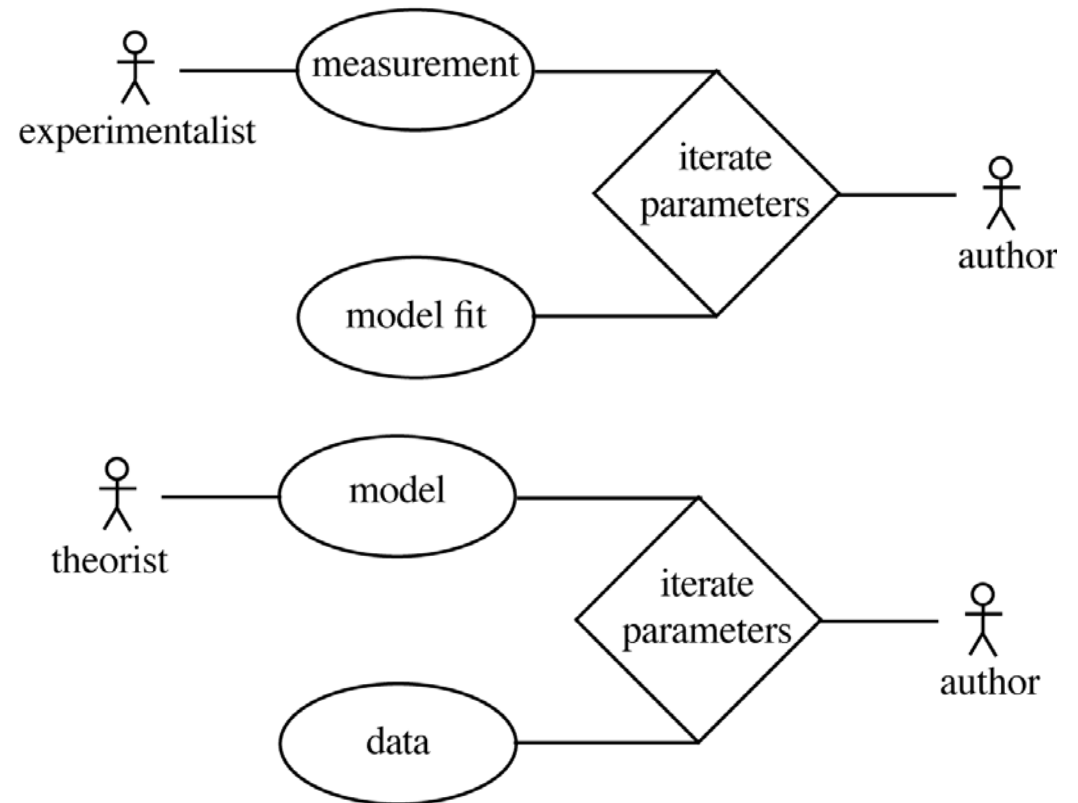
DANSE Project: Big Picture

- Enable new neutron scattering science by scientific computing.
- Build enough important and quality pieces to make DANSE the starting point for future developers.
- Detailed WBS, development process, and Earned Value Management
- M\$ 12 over 5 years (Anniversary June 1, 2006)
- Some support for SNS early instruments
- Releases to friendly users (3 subgroups now, other 2 soon)

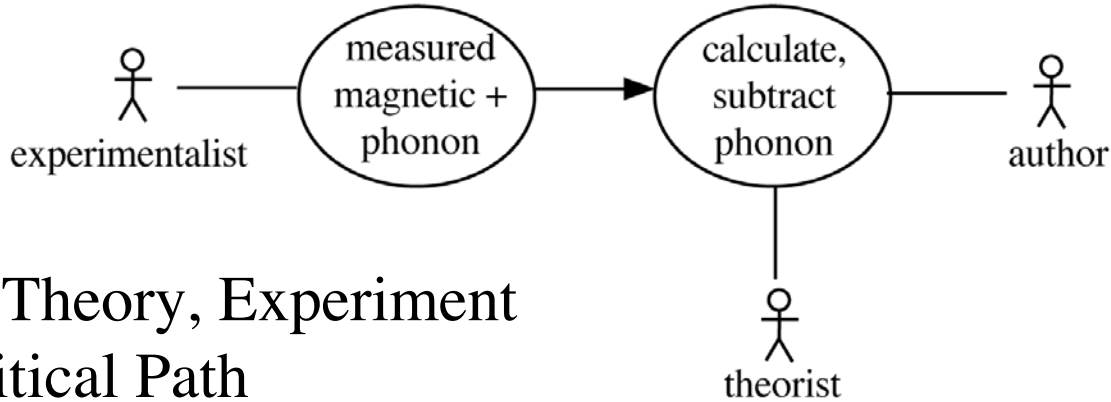


Dominant style of today

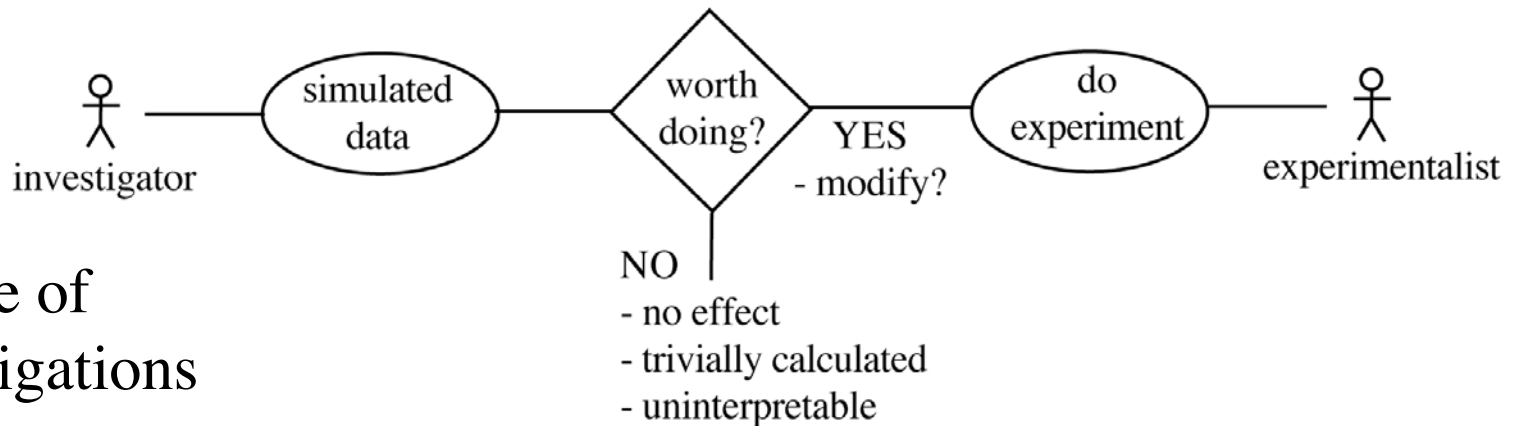
Natural Extensions



Near Future



Both Theory, Experiment
in Critical Path

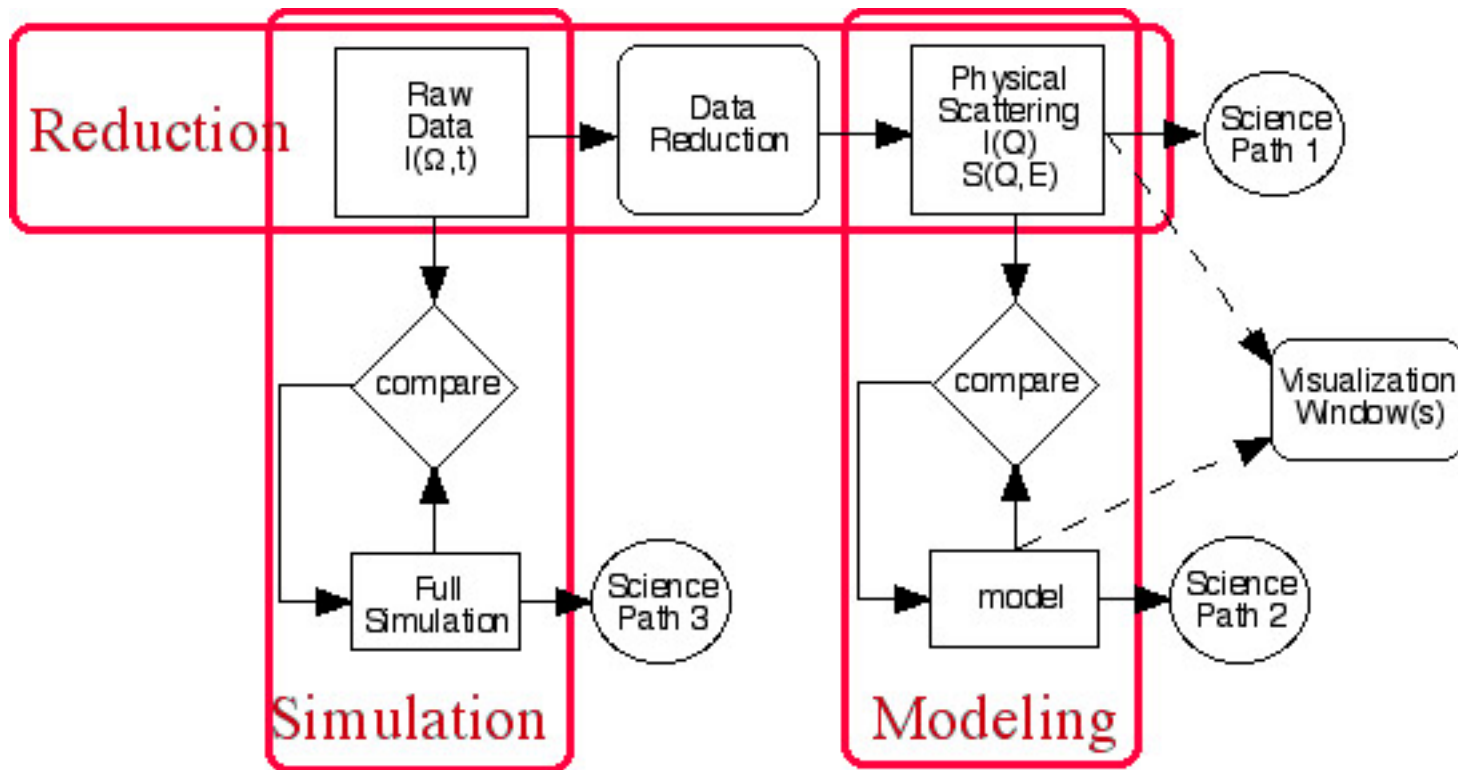


Choice of
Investigations

*Develop computing methods
for new types of discovery in
neutron scattering science.*

*Ensure a career path for scientists who add
value at this higher level.*

Three Paths to Discovery



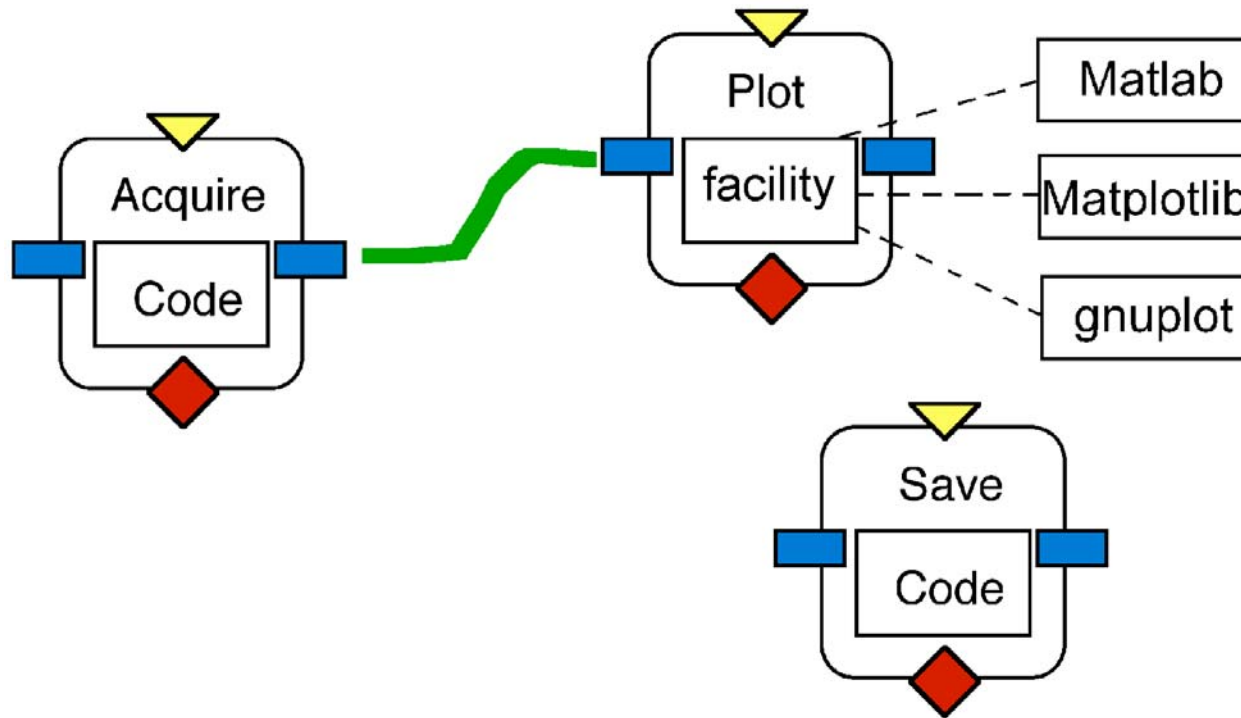
Features of the Development Process

- Misperceptions of specifications are the biggest source of risk in software projects.
- Subgroup leaders know what they want, our developers write specs.
- Good specifications and requirements up front
(revisited as appropriate -- every 6 months so far).
- Reviews, audits, and QA practices further minimize risk.
- Adapt commercial practices for scientific software development.

“Build to an Interface”

- Modular components with careful interface design
- Project-wide consistency and re-usability of components
- Swap out component core later with less disruption
- “inventory” of pyre component
- Allows automatic generation of GUI menus

Component Framework



Encapsulate science code within component.

User (actor) accesses core code through framework.

A component inherits methods from the framework.

DANSE Effort Today: Management

- Infrastructure tools are working (svn, trac, build)
- Development processes are in use
- Release management is partly working
- Earned value management is running
- Change, risk, configuration control plans are in place

Reviews (planning, design, code, release)

- Reviews shall help developers.
- Beware: 200 components \times 4 reviews = 10^3 reviews
- Code audits by Project Manager
- Priorities for Reviews:
 - Any Flagship Application
 - A component interfacing to many other components
 - Each new developer to do a review early

attendees: linjiao, mmckerns, swain, delaire, patrickh

JL> We can describe histogram as a mapping from a rectangular area in phase space to R^2 . Of course, we need an axis that is discretized into bins, and also R^2 represents the 2-tuple of data and error.

PH> So does that mapping do interpolation?

JL> No.

PH> What if asks for point that is not part of defining list?

JL> It throws an exception.

PH> What if x is floating point, and you have a precision problem, is it smart to throw a exception?

JL> That's an implementation detail that we must talk about with `DiscretizedAxis` and `GenuineDiscretizedAxis`. If a `GenuineDiscreteAxis` has ticks of floating numbers, than I should not allow a rounding-error. If we are talking a bout `DiscretizedAxis` with ticks only (on bins), then the rounding error should be allowed. For an axis with bins, then we should return a value whenever the input value is inside one of the bins.

PH> Where do you put interpolation?

JL> I do not put in interpolation... if you want to do interpolation of

WBS and Earned Value Management System

- WBS is regularly updated
- Subproject Statements of Work and Budgets are derived from the WBS every 6 months
- Earned value tracks the WBS
 - tasks (WBS level 4)
 - actual costs: from university accounting systems
 - earned value: percent complete assessed by:
 - development milestones, subproject leader, project manager
 - highlights from subproject → PM → PI → NSF
 - monthly

EVM Data on DANSE

Please inquire to Brent Fultz
for access to information
contained on this slide.

DANSE Project Execution Plan (draft)

DANSE functionalities

Milestones

Reviews

Management structure

Project Baseline

Project contingency

EVMS

Release Management

QA

SNS transition

License

Components

Flagship Applications

Table of Contents

1	Introduction	3
2	Mission Need	3
3	Project Description	3
3.1	Technical Description	3
3.2	Operational Goals	5
3.3	Key Functionalities	5
3.4	Project Milestones	6
3.5	Project Reviews	6
4	Resource Requirements	7
4.1	Budget Periods	7
4.2	Disbursements of Funds	7
5	Management Structure	8
5.1	NSF Division of Materials Research: IMR-MIP Program	8
5.2	Caltech: Principal Investigator	9
5.3	Software Architect	9
5.4	SNS Liaison	9
5.5	Project Manager	9
5.6	Interface Between DANSE and the SNS	9
5.7	Interface Between Caltech and the Subprojects	10
5.8	Interface Between the DANSE Project and the Neutron Scattering Community	11
6	Project Baseline	11
6.1	Acceptance Criterion	11
6.2	Work Breakdown Structure	12
6.3	Project Schedule	14
7	Project Controls	14
7.1	Project Baseline and Estimates of Baseline at Completion	14
7.2	Management of Contingency	15
7.3	Change Control	16
7.4	Earned Value Management System	17
7.5	DANSE Monthly Reporting	18
7.6	Earned Value Data	18
7.7	Management of Risk and Configuration	19
8	Release Management Plan	21
8.1	Inception and Design Assessment	21
8.2	Elaboration Phase	22
8.3	Construction Phase	22
8.4	Transitional Phase	22
8.5	Application Release Process	22
8.6	Release Criteria	23
9	Quality Assurance	23
10	Environment, Safety, and Health	23
11	Transition to SNS Operations	24

Developers' Infrastructure: Subversion and Trac

- Subversion Repositories
 - version control of files, directories, symbolic links
 - interoperable repositories, now approaching 100 for DANSE
- Trac ticket system
 - wiki and issue tracking system for software projects
 - integrated with Subversion
 - adapted to project management and auditing

Safari File Edit View History Bookmarks Window Help
 http://danse.us/admin/activity
 http://danse.us/admin/activity Google
 Truthdig MyLATimes Yahoo!News NewYorker The Huffington Post latimes Yahoo! Mail washingtonpost News (1290) NY Times BBC

DANSE Development Activity

Last 7 Days

Who	Repo-Rev	Commit Comment
Today		
btf	ARCSBook-140	Figures for new section on simultaneous multiple scattering and multiphonon scattering.
btf	ARCSBook-139	Added section on simultaneous multiple scattering and multiphonon scattering, following Q,E independence algorithm of Kresch. Essentially the same as described in Kresch, et al., PRB 75, 104301 (2007).
linjiao	buildInelast-15	added reduction-RHELserver5/install-deps/install-pylons.sh: script to install pylons
sylee	engdiffraction-101	design pattern code examples
mathieu	sans-255	Updated toy model viewer
mathieu	sans-254	more explicit error message...
mathieu	sans-253	update unit tests
mathieu	sans-252	Updated setup.py
mathieu	sans-251	Updated Doxyfile
mathieu	sans-250	Removed prototype files
mathieu	sans-249	Removed prototype files
linjiao	buildInelast-14	reduction-RHELserver5/start-webserver: script to start the server reduction-RHELserver5/install-deps/install-TeX script to install latex
mathieu	sans-248	Got rid of useless model
mathieu	sans-247	Added mini-tutorial for Doxygen front page
linjiao	measurement-1217	started user guide for "measurement" class
linjiao	histogram-1252	- added method "transpose" to Histogram class - improved Histogram GUI application so that pylab commands can be used easily - added user guide on how to use Histogram GUI
linjiao	pyregui-45	luban/gml/wx/HistogramPlotPanel.py: removed "toolbar", which seems to be not very useful and troublesome.
linjiao	buildInelast-13	added ARCS software website to the releaser.
Yesterday		
linjiao	buildInelast-12	reduction-RHELserver5: added "mpich2" to "install-deps". added script to build with mm build procedure.
jingz1	sans-246	add transformation through rotation matrix to Point3D
jingz1	sans-245	minor changes to pass windows compiling of ComplexModel(merge PDB & realspace model) class.


 Search

[Login](#) | [Settings](#) | [Help/Guide](#) | [About Trac](#)

Wiki | Timeline | Roadmap | Browse Source | **View Tickets** | New Ticket | Search

[Available Reports](#) | [Custom Query](#)

{1} Active Tickets (70 matches)

- List all active tickets by priority.
- Color each row based on priority.
- If a ticket has been accepted, a '*' is appended after the owner's name

Ticket	Summary	Component	Version	Milestone	Type	Owner	Cre
#182	Complete description of scripting task on wiki	admin		4.3.1:inception	task	aivazis *	04/
#172	clarify software production requirements	admin			task	mmckerns	04/
#170	Non-functional requirements need clarification.	boxmin		5.4.2.1:inception	defect	kienzle	04/
#169	Identification of acceptance criteria	boxmin		5.4.2.1:inception	defect	kienzle	04/
#168	Identify task timeline	boxm					
#167	Identify risks and risk mitigation plan.	boxm					
#166	Identify the system boundary, actors, and interfaces within each application.	boxm					
#165	Provide a clear definition of task scope.	boxm					
#164	Name target testers	boxm					
#163	Typo in pyre file FileLockingNT.py	Devel					
#162	Crystal Structure Class	simul					
#161	multiple histograms with same name cause trouble	histog					
#159	histogram review #3 action item	histog					
#158	Action items for Histogram Design Review # 2	histog					
#157	histogram design review #1 action items	histog					
#156	Complete 9.1.1 "Data Transformations"	reflec					
#155	Complete 7.2.3 "Inverse Problem Analysis"	engdi					
#154	Complete 6.2.3 "Model Independent Peak Fitting"	diffra					
#153	Complete 5.3.1.2 "McStas"	mcsta					
#152	Complete 3.2.2 "Build System"	reduc					
#149	supplementary specifications for reduction package	reduc					
#148	use cases for reduction package	reduc					
#147	risk mitigation plan for reduction package	reduc					
#146	roles and drivers for reduction package	reduc					
#145	clarify requirements for reduction package	reduc					
#144	define scope for reduction package	reduc					
#141	identify roles and drivers for instrument apps	instru					
#138	clarify requirements for measurement	meas					
#137	Define use cases for measurement	meas					
#135	Complete 5.?.?.? "Geometry"	geom					
#134	Complete 5.?.?.? "Reduction"	reduc					
#133	Complete 5.?.?.? "Instrument"	instru					
#132	Complete 5.?.?.? "Measurement"	meas					
#131	Complete 5.?.?.? "Histogram"	histog					
#130	suggestions for INS reduction plotting	reduc					

<http://danse.us/trac/tickets/report>



Ticket #57 (enhancement)

Reduction -- Working with funky detectors.

Status: new

Reported by: max Assigned to: linjl
 Priority: major Milestone:
 Component: reduction Version:
 Keywords: reduction Cc: mk

In no particular order:

1) Generally, we use the vanadium data and some knowledge of the energy of pressure of the detectors to try and correct for their not all being created possibly add to this the shading of some of the detectors from the way that that's what does it?)

2) Currently, it's very difficult to pick bad detectors to be removed from the data. In fact, it's often difficult to know which detectors are bad until the data has therefore propose the following:

Once the data has been reduced, you open an intensity plot of spe.pkl, an phi space that appears to have been marred by a bad detector. This is used detectors, which may then be displayed to the user -- possibly in a multiple (six to f) (dot to f) and (dot six). He may then pick the culprits out -- citi



Create New Ticket

Your email or username:

Short summary:

Type:

Full description (you may use [WikiFormatting](#) here):

B I A

Ticket Properties

Priority:
 Component:
 Keywords:
 Cc:

[Preview](#) [Submit ticket](#)

DANSE Effort Today: Personnel

- All senior personnel together since early 2003
- Professional staff
 - managed carry-over of personnel from CED award
 - new personnel in place, or coming soon
- Postdoc Software Developers
 - some delays in starting
 - full staffing is nearly achieved
- Graduate Students
 - can leverage DANSE in thesis research
 - some development work, much testing
- Undergraduate Students
 - mostly in computer science
- External collaborators

DANSE Effort Today: Technical

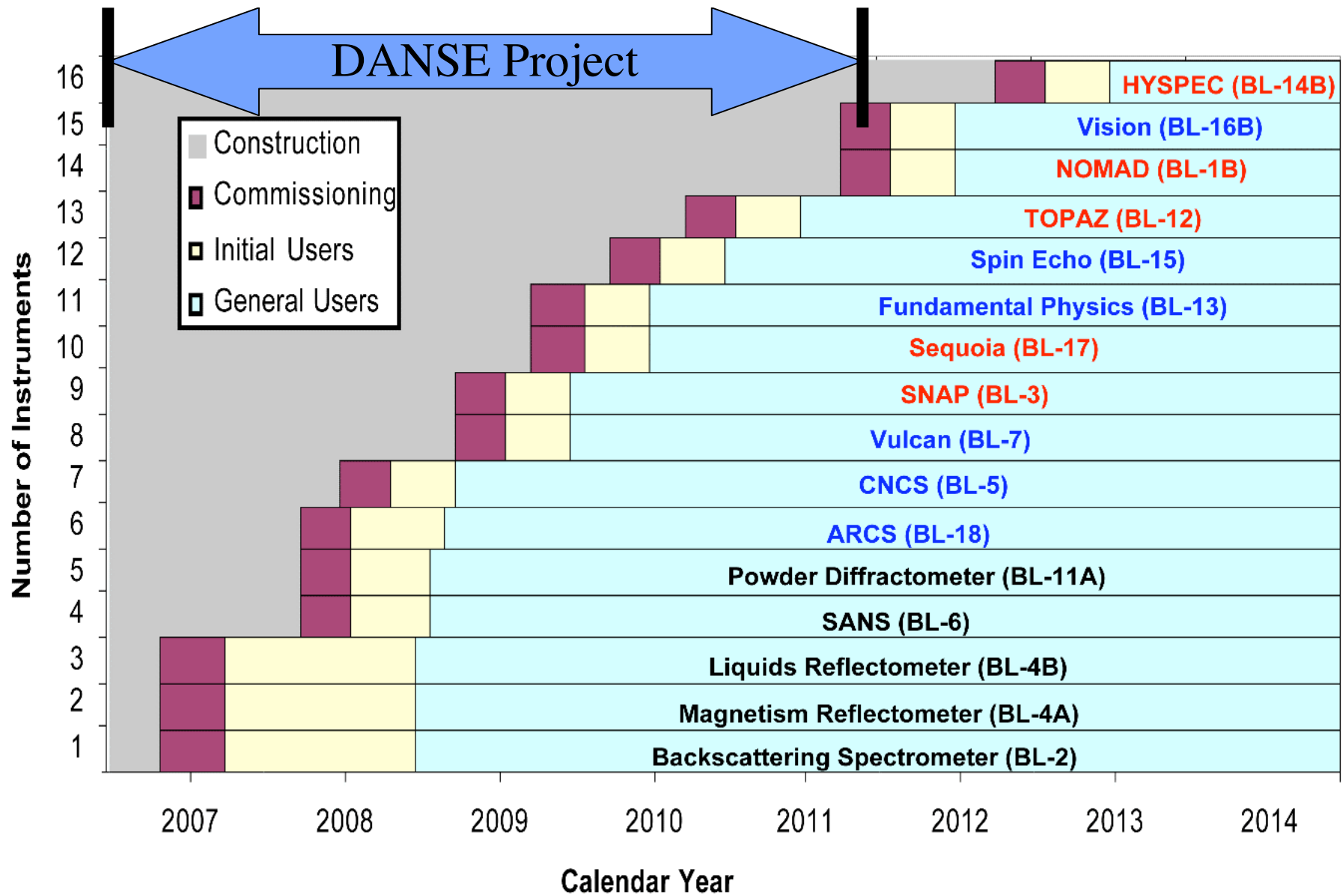
- Science requirements are stable
- Flagship applications are selected (see DANSE PEP)
- Common components for I/O, interface, and numerics
- Software development process is running
- Testing plans are under development
- Releases:
 - ARCS inelastic software in production, pyre framework
 - Reflectometry software in production, moving to framework
 - Diffraction package PDFgui
 - SANS package soon

Planning of Releases

- Chaos and pain are unavoidable in the first release
- Possible to hold to long-term schedule
 - Components can be added or deleted quickly
 - Development freeze for release 3-4 weeks in advance
- Regular revisions after first release

DANSE and the SNS

- DANSE will facilitate new science from the SNS.
- Good communication between SNS and DANSE
- Complementarity of software efforts
 - DANSE emphasis on advanced data analysis (modeling and simulation)
 - SNS emphasis on data services (acquisition, curation, reduction)
- Engineering consistency: SQRL consults for the SNS, subcontract with DANSE



Education

- Plan for pre-service teacher education at ISU (NSF RET)
 - develop K-12 lesson plans using content from subprojects
 - adoption by young teachers
- Nanoscience program at MSU
- Student involvement
 - minority student funds
 - testers of software and documentation
 - some in computer science
- Textbooks
 - Fultz and Howe, 3rd edn. (Springer planning publication now)
 - Experimental Inelastic Neutron Scattering (SNS distribution?)

Outreach to Broader Community

- Open source (BSD license) facilitates collaborations and interactions

SNS ASG, NeXus/NIAC, McStas, NumPy, SciPy, cctbx, NAMD, VMD, MMTK, gnuplot, ISAW, matplotlib, CEDS Berkeley component

- Workshops and Organizations

ISAW collaboration (Mikkelsons), M. McKerns to Australian Research Council NCRIS [National Collaborative Research Infrastructure Strategy], XSD Software Workshop, Paul Kienzle elected DANSE representative to NIAC, Jiro Suzuki and Tony Lam are here.

Broader Impact

- Algorithm Development Moves to Higher Levels
 - flexible rearrangements of DANSE components
 - subproject science applications
- Demonstrate a New Approach to Scientific Software Development
 - modular approach with O-O design
 - some interest from computational scientists
- Intense Interest in DANSE
 - international neutron facilities, ASII proposal
 - synchrotron radiation research
 - other user communities

Challenges and Risks

- Optimizers for modeling -- we don't understand the math.
- Microsoft Windows (builds, user expectations, multiple configurations)
- User science expertise sometimes needs to be high.
- Balance between perfect and realistic development practice.
- Managing change.
- How much testing can we afford?
- How will we support friendly users?

Summary

- Lots of low-lying fruit for computation and neutron science.
- The DANSE Project is initiated, and path forward is clear.
- Roles of DANSE and SNS have evolved naturally:
 - SNS software affects the quality of the data from the facility.
 - DANSE does data analysis at a higher level of abstraction.
- Integration of Design, Development, QA, Release Management.
- Subgroups will present details of progress and plans.