Requirements gathering and user engagement

Discussion Lead: John Rehr

Scribe: Jay Alameda

Participants:

- Jay Alameda
- Richard Brower
- John Cazes
- Tim Cockerill
- Jodi Hadden
- Chathura Herath
- Brandon Hill
- Gideon Juve
- Murat Keceli
- Robert Kosara
- Suresh Marru
- Manish Parashar
- Abani Patra
- Shawn Shadden
- Sarom Sok
- Ross Walker
- Vineet Yadav
- Elias Balaras

Questions:

Requirements Gathering/User Engagement

- What are the mechanisms used today to gather and understand user requirements? How effective are these mechanisms?
  - have a couple of thousand users in John's community, hear from them in email, forums, telephone calls
  - Ross - finds user requirements dont' often meet with reality
    - what users would like in terms of performance, reliability are far beyond what we can deliver
    - would like to run at 10ns/day on newest supercomputer, can't do this due to limitations - they try to do it, and it doesn't work, and then will come back - seg fault on 10k gpus, then complain about this -
    - these are things that are impossible, but a lot of users would like to do this!
  - would like to talk about what we can do to make facilities more useable today
  - cynthia - still a new sw developer, still in the startup phase -what she has found to be really useful, is embed with some users, and the way she does this, go around on college campus, find people with genomics problems, not effectively solving them, and work with them from start to finish on a project that represents what std user of sw would have to do - that is a wonderful 'from the beginning' approach. really good to have non-computer savvy local user
    - do you have workshops?
    - not yet, hopefully we'll be in the place where we'll be solicitating collaborations, moving to workshops
  - kevin - 4 ways
    - email questions - threshold, though, only get folks with sufficient self-confidence
    - workshops - where people are taught to use code in hands-on way
    - mailing lists - if they have a critical mass, one code has this - dozen of regulars that take the time to reply and help people - share expertise, this is a large factor in success of the code
    - giving talks at conferences, reaching general audience, not just computer science specialists - really like material science conferences, get good feedback heere
  - Brandon - this doesn't scale, but user interviews - looking at R and questioning users on what they use, what they don't and why - we have certain things we are wondering about - find it is productive to ask users about long-running codes, ask about their experience with tool,
    - ask them with a bright light, dark room :)
    - network through connections, find users running long-running codes - offer to profile their codes, this helps with interest to talk to us -
  - distinguish between requirements gathering for scientific sw, versus requirements gathering for middleware (eg nanohub, science gateways) - for science gateways, his experience is that
    - people try to impose upon these computer science codes certain ideas, hard to get people to adapt to new code features
    - sit with scientists, try to understand their science - eg, sit with users of gridchem, figure out what they want to do - figure out what we can to do improve useability of the system -
    - we are in an unique position, as we are doing xsede gateways, they come to us, so help them out -
  - In John's community, big national facilities, synchrotron - where use the sw to analyze expts, they are the bulk of the user group, this user group has a life of its own - when they find problems with the code, send email or something like that - not sure if there is another way to engage -
  - Ross - has a survey going, co-running with nvidia, encourage users to write about the problems are solving, wishlist, so can prioritize
    - bribe the users, nvidia has several c2050 gpus to give away at the end (random drawing) -
    - no idea of what percenatge of users would take time to fill out survey
  - John - have demo site where try out new features of site, this is very early -
  - Ross - have md cluster program, get a login on manufacturer's machine, and try out the machine running md code - this seems to work well
  - usage statistics -
    - sw that phones home would be cool...
    - John Cazes, tracks things about users, where check out what modules are loaded
    - NICS - keeps db of sw -
    - what percentage of users use installed executable -

      - amber - always install his own, to know what patch level it is at -
    - John - guessing, maybe 20-25% use this
      - libraries - are much higher - nobody wants to build petsc -
      - we update the sw when users tell us about new patch level, or request to go back -
    - Ross used to update Amber at all the sites -
      - John could check into funding -
    - tweaks you have to put in, are already done at the center,
    - commercial sw tries to report to vendor everytime something goes wrong, how hard would this be
    - there are a number of sw engineering packages
    - microsoft has an automated system that looks at
    - even if it is publically known, users don't like phone home -
      - depends on what is reported
      - maybe not an absolute no-no under all circumstances
    - scientific sw - is different from productivity sw - if solving a problem, then happy to get help with this, different from operating system with commercial organization
      - maybe build in compile options - to not have reporting, maybe in Europeon countries -
      - could have incomplete stats -
    - Citations - Ross - how we measure most people's success - is citations, a good way to measure success
      - amber - has a publication, which 95% of amber users cite - this works for amber
      - what about middleware?
      - maybe need a better way to cite sw -
    - John Cazes - has a lot of commercial users, have licensing information, not citations -
    - Cynthia - review grant proposals, tenure cases, etc in bioinformatics, see all the different ways to represent how sw is used - sometimes, "just a tool I use" - vis sw is one of the most abused sw in bioinformatics, as don't tell how made a figure - what see people do is track downloads, this gets you into ballbark - or sign an academic license agreement - see this sometimes in tenure apps -
      - downloads
      - contacts regarding sw
      - citations
      - envision users as anyone that ever downloaded sw - and use this to get a ballpark
      - this may be the highwater
  - We can use surveys, which are reliable to the extent that self-reports are honest. The problem with this is a lack of incentive. (dc)
  - Informal interviews with software users (e.g. people we already know who use the software) can shed light on user requirements and preferences, but these can be skewed by social desirability (desire to appear in a positive light to the interviewer). (dc)
  - Usage statistics functionality built in to the software. For any psychological study, these require some form of consent from the user, so we limit ourselves to the sample of people who are willing to provide the usage statistics. (dc)
  - Though it may not be a common practice for gathering user requirements, we could look on forums/message boards about user-to-user communications about the software--good for users who are anxious about communicating with the developers. (dc)
  - Pegasus team interacts directly with users. Since Pegasus is middleware, it is occasionally difficult to justify implementing features that will be used by only one user. Some users ask for features that are not generic and fit only their use-case/application. Need to work with users and identify their fundamental requirements, not just the wish list of superficial features that would help their application.
- What is user engagement and how do we encourage it?
  - John - brought this up with people on XSEDE, do we do user surveys, would like to add some questions on community codes - what are the incentives to complete survey, can we give them hours? bribe with best buy cards in TG -
    - user surveys is an ongoing user engagement
  - do we do enough?
    - survey writing is an art of its own - XSEDE is hw and service oriented, don't want to ask detailed hw questions - they are users - they shouldn't be expected to know this - need to ask questions that get to the heart of this -
    - takes survey after survey to see the results
      - need to advertise the surveys too -
  - is there adequate incentive for us developers to do this engagement with users?
    - current codes not user friendly, this is lamentable -
    - if you enage users, they may ask you to do something for them .... :)
    - how do we design this to get beyond the groups we are working with -
    - at first engaged users is wonderful, they help you, and then becomes a huge pain, so many want so much from you -
    - if you have engaged users, do you have resources to support that level of users?
  - John likes this set of questions, should be getting feedback from userrs - should be part of the best user practices -
  -
- Are there any potential benefits from working with the Extreme Digital requirements gathering/management efforts?
  - if you think about -
    - cycle of process - talking about requirements gathering, this is step one - this is step one - data collection
    - how do you deal with the data - is it going to sit in emails, or list of notes on your desk, or...
      - collect/categorize/sort this information - with XSEDE project, we use product from Rational "DOORS" - requirements collection and management sw - collect and work with requirements, need to analyze those and make decisions
    - few specific examples - on information collects?
      - this is a repository, collects whatever you put in it
      - categorize input, find commonalities - decide if what you collect from users - other themes - is speed of data transfer a common recurring theme -
      - convene a group of people to review reports from the collection, then do some more investigations
      - and then - decisions on whether will respond to the need or not -
      - we'll collect information from users on what is slowing down their work, for instance
        - long queue wait times
        - large files that need to be moved periodically
      - take this information, stuff into db, and use as a tool to see how progress the requirements into actions -
      - all this information comes in - knowing what users say is one problem, knowing what to do with it -is another -
    - maybe have a website or link - that we can look at?
    - promised sharing information from this -
  - Brandon - thinks this is critical - see common problems, on things that can change that may impact multiple problems, would be good to have feedback to users - on what multiple users are saying - can see many users that have the same problem -

- getting these things more open and shared would be very useful, get others thinking about it -

- Are there any other questions not mentioned here?
  - add to the nuance of sharing - for a long time, we didn't work together - teragrid, xsede, came about 6-7 years ago, program came about in 2001, this started with a bunch of institutions that competed with each other to build next build supercomputer -
    - sharing what users tell you is their next great need is scary -giving up competitive advantage -
  - prior to this, the centers would work together until a solicitation comes out, then clam up, ...
  - this comes back to community frameworks, getting people to work together - coopetition thing, where supposed to cooperate with each other, but compete at other times - takes time to work this -
    - this is not easy, the social aspect is the hardest part - took a while to do this in xsede, taking better care of users now than as centers
- how does a community prioritize their user needs -
  - Brandon - turn this - is anyone doing this?
    - if have one user with a complaint that is doing bigger science than any other, do we listen to them?
    - are all voices equal?
    - some users - write persistent emails to you -
  - John - sometimes say we'll put on list, sometimes if workign on actively, then give higher priority -
- in open source model, requests captured somewhere, when do release, say that this is high priority, block, etc - community consensus -
  - if sw is mainly driven by one fat client, and a few other thin clients, then fat client drives the priority
- Manish - if a community code, have a way to prioritize which feature -
  - Surveys that get done, does anyone do things like publications that try to assess how big of a cat you are?
  - John - don't ask these sorts of questions, but know them -
  - much broader survey - lose a lot of information in the broad survey -
- have to be careful with how many questions you ask - how much personal information can you extract while getting to the heart of the technical questoins
  - and do you want to perpetuate the common perception, money attracts more money - do you want to tie to grant pot, would be really careful about putting this in that way -
  - does anyone conduct user survey when developing new sw or new module -
  - usually the other way around with us, write the new module, finally to the top of the priority list -
- one question - do we ever ask users how user friendly our sw is -
  - if we do, we'd get a lot of feedback - what do users really want, that they are not getting -
  - what woudl be transformative, sw that works like your apple computer
  - can't expect miracles, but in a lot of cases things can be a lot cleaner - one thing that we coul
  - (comment about IO)
    - could use HDF5 -
  - this is a big open-ended problem - see this where too many parameters, too flexible, becomes confusing - where do you draw the line -
  - release early, release often - rather than wait too long - surprise with new interface
    - and different levels, novice to expert
    - for novice, want it to be easy and
  - this is a dangerous question - could be talking about apis, or mouse click to take care of it -
  - may be something that put into survey, catch-all kind of thing, reveals some of the dissatisfaction
  - dangerous: afraid to ask???
    - have to know your mission - may not want to be makign something mouse click driven?
    - so if they want something mouse-driven but it is not on your path, then can suggest to look elsewhere
  - Ross - can be nice to have energy barrier to some of these things - could make point and click -but don't want Amber to be point and click, as skeptical that users that could do this -
  - complexity of doing simulations of proteins, so many opportunities to - so many parameters to tweak
  - Kevin - think this is an argument where this is truth in certain cases - but, if you are at edges of computatinoal science, need a bunch of opaque parameters - if you have some generic code calculating ground state of crystal, this should be black box -
    - In Density Functional Theory (DFT) - 100k different functions you could chose,
      - give them a default -
    - Cynthia - not offering users every last possible obscure sw packages, offering the packages to community,
    - this is the whole part of documentation and user friendly, your sw should educate people on process they are doing - many ways to do this in your interface
    - John - in his code, have default values suggested -
    - Abani - evolution - things we talked about 25 years ago, are no longer an issue - will things we are discussing today be pushed into hidden layers later?
    - Kevin - do you want to be in the driver seat -the better you can have users use sw, the better
    - Brandon, we are getting into tangent on good users of modeling and simulation - could model this with this program, complete wrong theory to do this - but - have a lot of new uneducated users coming in,
      - what about checks in the code
      - some of these things are difficult to detect
      - parameters - neat acids well parameterized, well understood - but what about ligands -
        - used to have to pick parameters by hand
        - now have sw that helps you but not foolproof
        - how to check - someone gets a binding energy that doesn't hold up in expt - can go back through and figure out
        - can you detect it in the code? no -
        - not weird numbers, you will get -3k Kcals, rather than -4K Kcals -
        - or you can overwhelm users with "you are doing things weird" messages -
        - in DFT, error cancellation can occur, if compounds contain dispersion interactions, you don't want to use traditional DFT functionals (LDA, GGA)
        - could conceive to write expert system, that takes human intuition - this is a different project then
        - simple checks are good
        - Vineet - spend most of your time in parameterization, can't describe the parameters as defaults - as you go
        - users of sw - should understand the underlying scientific theory (Ross), Vineet agrees -
          - should know what Hartree fauk is - if using Gaussian, for instance -
          - need decent undergraduate course education, at the least -
        - this is what we want to have as our user base
      - need to put checks and balance in the sw to help users -

- cynthia - what about the burden of needing to know how ncbi blast - blast algorithm, amazing the diversity of understanding -
  - for a phd defense, sure, but for a biologist?
- this is a systematic problem that we can't have a solution for, we can't say the phd is not the new bachelors -
  - we won't get there -
- another approach - there will be idiot calculations done with your code - people will need to learn about your code -
  - if you have active mailing list, may get a lot of novices checking in with their results -
  - still have peer reviewed publication system, which should weed out the rubbish -
  - greater ease of reproducibility - peer review can catch -this is what trying to do in Cynthia's sw - not doing it in their publications -
  - other way in peer review, list parameters necessary to reproduce the calucation, code should have log file to log the parameters
  - Cynthia - in general sense, maybe the purpose of the sw centers should be to help develop more respect for sw as thing you need to understand -
  - education: should be on the priority list, how much work goes into producing sw -
  - this isn't appreciated widely at all
  - Abani - this is community dependent -
    - numerical linear algebra has some high standards
    - elsewhere - how do you encourage this?
    - maybe this is the kind of thing we should advocate?
    - John supports giving enough information to replicate -
  - Ross - someone runs 2-3 simulations, on supercomputer X, look at results, draw some conclusions, this is what write up, a few TB that can't be stored - delete this, and maybe keep input files -
    - no way given newton's equations of motion, that can exactly reproduce
    - can simulate an ensemble - conclude that every paper published is uncoverged
    - reproducing the conclusions - which can be simple -
    - use pdb blah, force field blah, and saw this interaction with this ligand
      - this isn't so bad
      - but sometimes, may use an edited forcefield - and don't know it -
  - simulation is still supposed to be science - conclusion could still be that computation is wring
  - what do you need to communicate in a research paper - maybe no longer just words, but also data structure - you must submit input files -
    - is there a way to capture how we've done computational expt that is as detailed as your lab notebook -
    - having something on paper or electrons gives you a basis - this may be a larger communicating about computatonal
      - journals are doing more about this these days, but there are limits
    - John had a graduate student, anything that couldn't reproduce is "art" -
  -

Summary:

Contact with users:

- Forums
- Surveys - incentives
- embedding/direct collaboration with users
- workshops at conferences
- email lists
- "phone home" capability in software
  - problem
  - usage statistics
- download statistics

User Enagement

- not clear we do enough, this sort of effort should be part of best user practices - to encourage best user engagement.

Potential Benefits from working with XSEDE User Requirements and management capabilities?

- not just about gathering requirements
  - "no data collection without analysis
  - no analysis without decision
  - no decision without action"

User friendly nature of software, building in checks

- mixed conclusions on feasibility
- cross-disciplines - may need to do computations outside your field -
- phd: gives you appreciation for what you don't know - and understanding of methods involved
- this goes to appreciation of the software, this is an education thing - learn how hard it is to do various computations -
-

Notes: