Case Studies in Research Software Sustainability

Rajiv Ramnath
(ramnath.6@osu.edu,
https://cse.osu.edu/people/ramnath.6)
Professor of Practice, Computer Science and Engineering
Co-Director, Smart & Connected COP, TDAI,
https://tdai.osu.edu

Daniel S. Katz
(d.katz@ieee.org, http://danielskatz.org,
@danielskatz)
Chief Scientist, NCSA
Research Associate Professor, CS, ECE, iSchool

Recap: Software is indispensable to research

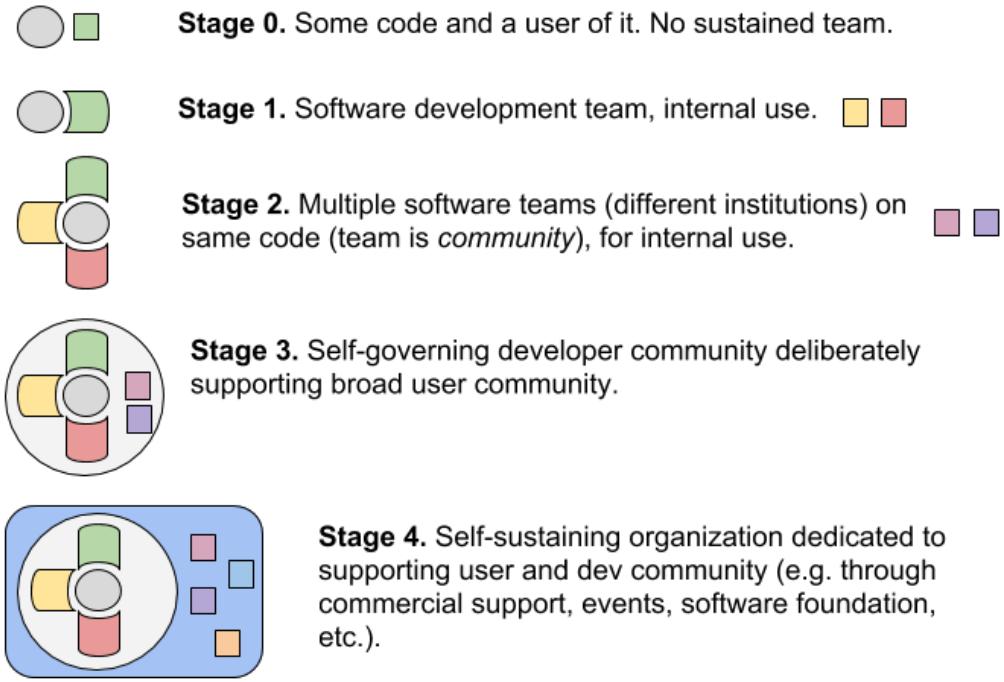
Recap: Research software is a stack

1. Project-specific software (developed by researchers): applications, scripts, workflows, computational notebooks, special-purpose libraries & utilities
2. Discipline-specific software (developed by developers & researchers): that implement disciplinary models & methods
3. Scientific infrastructure (developed by developers): middleware, libraries & utilities used for research in many disciplines
4. Non-scientific infrastructure (developed by developers): operating systems, compilers, and support code for I/O, user interfaces, etc.

Software builds & depends on software in all layers below it; any change below may cause collapse

Recap: Research Software Stages

Schematic stages of open community for research software



Stage 0. Some code and a user of it. No sustained team.

Stage 1. Software development team, internal use.

Stage 2. Multiple software teams (different institutions) on same code (team is *community*), for internal use.

Stage 3. Self-governing developer community deliberately supporting broad user community.

Stage 4. Self-sustaining organization dedicated to supporting user and dev community (e.g. through commercial support, events, software foundation, etc.).

S. P. Benthal, Software Incubator Workshop: A Synthesis, http://urssi.us/blog/2019/02/25/software-incubator-workshop-a-synthesis/

The Axes of Sustainability

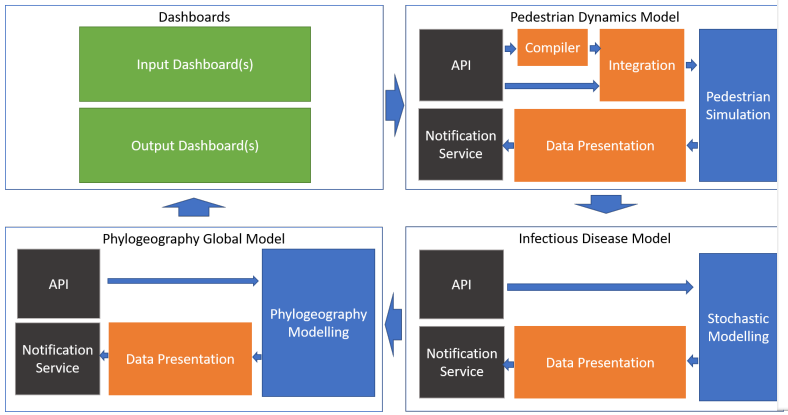
1. Intrinsic sustainability - reduces the amount of work needed to sustain
   - Developer training (carpentries) -> good software engineering practices -> usable, maintainable software
2. Sustainability of need – increases the motivation
   - End-user engagement -> Community development -> Endorsements and voluntary contributions
3. Sustainability of resources – pays for the effort, retention of expertise – the hard part!
   - Volunteered time
     - Motivated by: Interest, vested need, credit for contributions (citations, recognitions)
   - Institution supported (considered research or teaching CI, IDCs on grants)
   - Funding agencies supported (NSF, DOE, NIH, Foundations, Industry)
     - Types of funding: Direct funding, use fees, infrastructure “credits”
   - Commercialization
     - Licensing fees, industry gifts, support from commercial tiers

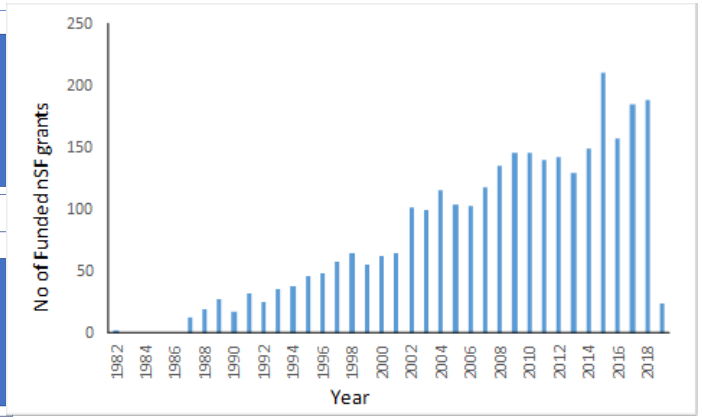
Case Study: VIPRA (PI: Ashok Srinivasan, UWF)

Description: COVID propagation (Project Specific) Pedestrian Dynamics (Discipline Specific)

Status: Stage 0 -> 1

1. Intrinsic: DSL, modular components, Chainbuilder workflow, TDD, continuous integration
2. Need: Growing contributors, highlighting value:
   - 2020 stakeholder workshop
   - COVID applicability
   - Local government
   - Papers, 200+ news articles
3. Resources: core team developer funded by federal grants





Year	Community Size					
	Pedestrian Dynamics		Infection modeling		Others	
	Contributors	Users	Contributors	Users	Contributors	Users
1	3	15	3	10	2	5
2	8	25	7	25	5	15
3	20	50	10	50	10	50

Case Study: Pegasus (PI: Ewa Deelman, USC)

Description: Scientific workflow manager that automates complex, multi-stage processing pipelines

Type: Scientific Infrastructure

Status: Stage 3

1. Intrinsic: Portable, modular stack, robust runtime, field tested, beta tested, standard dev. Tools (JIRA, git etc.)
2. Need: Established in astronomy, bioinformatics, earthquake science , gravitational wave physics, ocean science, limnology. Supporting large installations, Pegasus Users Group
3. Resources:
   - Contributors: Experts (ISI, HTCondor personnel)
   - Funding: NSF, DOE (R&D) -> DOE (pilot capabilities) -> NIH (deployment)

Download Metrics

Number of downloads	36,920

Top Planner Domains

Domain	Workflows	Tasks
isi.edu	77,820	85,556,479
atlas.aei.uni-hannover.de	27,200	21,684,075
ligo.caltech.edu	9,374	34,291,738
mps.mpg.de	8,120	24,360
eu-west-1.compute.amazonaws.com	4,823	12,327

Case Study: Parsl (PIs: Kyle Chard, Dan Katz, Illinois)

Description: A Python library for programs that use Python functions and calls to external executables
Status: Stage 2 -> 3

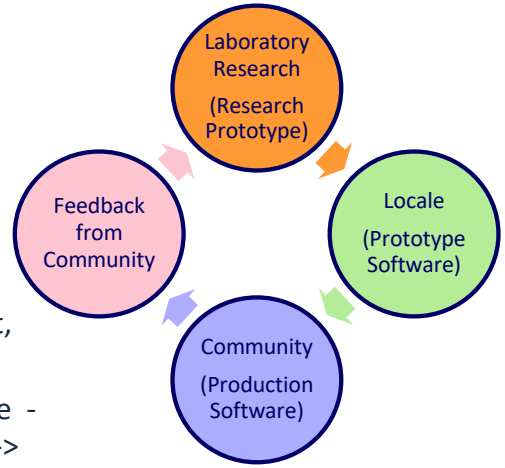
Type: Scientific Infrastructure

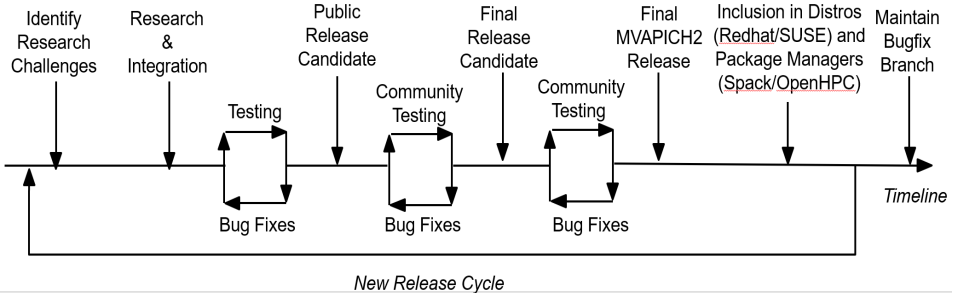
1. Intrinsic: Quality, commitment of team, general applicability, platform, standardized development process, automated testing, code review, continuous integration, standard tools (git, Travis.ci), Python features for type checking etc.
2. Need – increases the motivation
   - End-user engagement -> Community development -> Endorsements and voluntary contributions
   - 20-40 projects, 2000 downloads, Citations ~60, 5000 website visitors/month, 1000 documentation visitors/month, 180 slack members
   - ParslFest: 40 (2019) 60 (2020)
3. Resources
   - Contributors: 52 (10 internal)
   - Funding: NSF CSSI award for development, NSF CSSI project (funcX) that uses Parsl, Revenue from large and small projects that use Parsl, Code contributions (large) from external projects and (small) from individual users

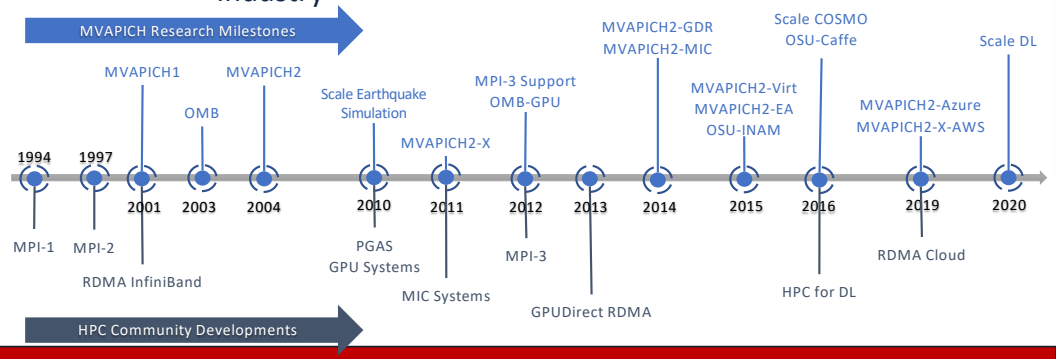
Case Study: MVAPICH (PI: DK Panda, OSU)

Description: MPI Library Stage: Stage 3 Type: Scientific Infrastructure

- Intrinsic
  - Value: Committed support, Shared addresses (RDMA), application and infrastructure (GPU, network) aware models, Innovation
  - Translation cycle: Lab -> Feedback -> Implementation and Deployment -> Research
- Need
  - Communities: TOP500 systems, National Labs, HPC centers, Industry
  - Earthquake simulation, DL frameworks
  - Publicity: Gordon Bell competition, Workshops (SC), MUG, Publications
  - Metrics: 20 years, 1,260,000 downloads, 3,125 organizations, 89 countries.
- Resources
  - Core team (PI, postdocs, sysadmin, graduate students)
  - Industry support (AWS, Intel, Mellanox, Microsoft, Nvidia)
  - Funding support: NSF core -> CSSI -> OBR, DOE, DoD -> Industry







https://doi.org/10.1016/j.jocs.2020.101208

Essential Elements of Sustainability

- Environmental incentives: Citations, recognitions, promotion, funding
- Core, committed team
- Quality: As robust, accessible as appropriate for user base and users
- Target: Science with high intellectual and societal impact.
  - VIPRA: COVID, Ebola …
  - Pegasus: Astronomy, bioinformatics, earthquake science , gravitational wave physics, ocean science, limnology …
- Dissemination, visibility: Access, user groups, showcases, publicity
- Vibrant community and partners (individual researchers, research communities, contributors, industry)
- Dependencies that are themselves sustained:
  - VIPRA: Chainbuilder
  - Pegasus: Python, Jupyter, Java, R, desktops, HPC centers, OSG, Clouds
  - MVAPICH: MPI standards, CUDA, ROCm, InfiniBand, Omni-Path, High-Speed Ethernet, Commercial hardware
  - Parsl: Globus

Credits

- Thanks to:
  - Colleagues and contributors: Vipin Chaudhary, Ewa Deelman, David Hudak, Dan Katz, DK Panda, Sushil Prasad, Ashok Srinivasan, Hari Subramoni, Karen Tomko
  - NSF: Where my interest in software sustainability began.