

HEP Software Foundation (HSF) Initiatives

Benedikt Hegner
(for the HSF Startup Team)

PH-SFT Group Meeting
30.11.2015

Motivation

- Much of our HEP software is now old (> 20 years) and needs to be adapted to more modern standards
- Paradigm-shift resulting from the evolution of CPUs
- Use of all resources available to our community such as HPC, commercial clouds, volunteer resources
- Must attract people with the required advanced skills and experience
- Ensure interoperability with software developed by other scientific communities
- Opportunity for sharing software between different experimental programs

Objectives

- Share expertise
- Raise awareness of existing software and solutions
- Catalyze new common projects
- Promote commonality and collaboration in new developments to make the most of limited resources
- Aid developers and users in creating, discovering, using and sustaining common software
- Support training career development for software and computing specialists
- Provide a framework for attracting effort and support to S&C common projects
- Provide a structure to set priorities and goals for the work
- Facilitate wider connections; while the HSF is a HEP community effort, it should be open enough to form the basis for collaboration with other sciences

HSF Timeline

- Jan 2014: [HEP software collaboration proposed](#)
- Apr 2014: [HEP software collaboration meeting](#)
- Spring/Summer 2014: gathering White Papers from the community.
- Oct 1 2014: [Startup plan approved and startup team established](#). Agreement communities and software domains to focus on initially.
- Nov 11 2014: [White Paper Analysis and Proposed Startup Plan](#) released, followed by discussions and contact meetings with many parts of the community prior to the SLAC workshop.
- Jan 20-22 2015: [SLAC HSF workshop](#) established concrete activities and next steps for the HSF.
- Apr 17, 2015: [HSF meeting at CHEP 2015, Okinawa](#) to present progress, assess opportunities emerging from CHEP, and discuss next steps.
- In addition a lot of hands-on work!
- June/July 2015: Intensive discussions in Packaging Working Group
- Sep 2015: [Technical Notes](#) policies published - more in the queue
- Sep 2015: HSF on WikiToLearn
- Oct 2015: Internal Evaluation of new Knowledge Base finished
- Nov 2015: [New website](#) online
- More in this presentation...

HSF Activities and Working Groups

Activity/Working Group	Objectives	Forum - Mailing list
Communication and information exchange	Address communication issues and building the knowledge base Technical notes	hep-sf-tech-forum
Training	Organization of training and education, learning from similar initiatives	hep-sf-training-wg
Software Packaging	Package building and deployment, runtime and virtual environments	hep-sf-packaging-wg
Software Licensing	Recommendation for HSF licence(s)	hep-sf-tech-forum
Software Projects	Define incubator and other project membership or association levels. Developing templates	hep-sf-tech-forum
Development tools and services	Access to build, test, integration services and development tools	hep-sf-tech-forum

Startup Team

- Amber Boehnlein (SLAC)
- Peter Elmer (Princeton)
- Daniel Elvira (FNAL)
- Frank Gaede (DESY)
- Benedikt Hegner (CERN)
- Michel Jouvin (LAL, IN2P3)
- Pere Mato (CERN, co-lead)
- Dario Menasce (INFN)
- Elizabeth Sexton-Kennedy (FNAL)
- Graeme Stewart (Glasgow)
- Craig Tull (LBNL)
- Andrea Valassi (CERN)
- Brett Viren (BNL)
- Torre Wenaus (BNL, co-lead)

More details, including meeting minutes, here:

<http://hepsoftwarefoundation.org/organization/team.html>

Communication and Information Exchange

- Collaboration lives from communication
- The team spent some effort on setting up various communication channels:
 - HSF [Website](#)
 - based on GitHub / Jekyll / Bootstrap
 - Mailing Lists (Forums)
 - using Google Groups
 - Knowledge Base
 - Technical Notes
 - Newsletter

Mailing Lists

- HSF Forum
 - <http://groups.google.com/d/forum/hep-sf-forum>
 - ~120 members
- HEP S&C community Forum
 - <http://groups.google.com/d/forum/hep-sw-comp>
 - ~350 people have signed up
 - General mailing list everybody in our field should subscribe to
- Other specialized lists
 - Training - [hep-sf-training-wg](#)
 - Packaging - [hep-sf-packaging-wg](#)
 - General HSF technical discussion forum - [hep-sf-tech-forum](#)
- Reminder: Google-free self-signup to lists
 - Simply send a mail (subject and content irrelevant, can be empty) to <listname>+subscribe@googlegroups.com, e.g. for the list above, [hep-sw-comp+subscribe@googlegroups.com](#)
- See the '[Get involved](#)' page on the website for details

Topical Forums

- **Concurrency Forum**
 - Technical issues to embrace concurrency in our software
 - Ongoing activity
- **Reconstruction Algorithms Forum**
 - All matters of event reconstruction and pattern recognition software
 - **First mini-workshop this week on Dec 3**
- **Machine Learning Forum**
 - ML discussions and code development in the context of HEP
 - Development of relevant tools, methodology and applications

- **Software catalog, software categories, science fields, community, and events**
 - implementation is a browser-based app (javascript client, node.js server, json in between, MySQL)
 - authentication is via github, google, facebook etc.
 - **emphasizes easy adding/editing of content, and extensibility. Adding content should be fun.**
- Available at <http://hepsoftware.org>
 - Comments/feedback are welcome!
 - Just start filling it!
- Based on similar software inside ATLAS

Experiments

- Experiments & software
- Software & experiments
- ALICE
- Alpha Magnetic Spectrometer (AMS)
- ATLAS**
- Belle II
- BES III
- CAPTAIN
- CDF
- CMS
- COMPASS
- Cuore Experiment
- D0
- Dark Energy Survey (DES)
- Daya Bay
- DUNE
- FAIR
- Fermi Gamma-ray Space Telescope (formerly GLAST)
- HARP (PS214) - The Hadron Production Experiment at the PS
- Heavy photon search
- KOTO
- Large Synoptic Survey Telescope (LSST)
- LArIAT
- LHCb
- LSST Dark Energy Science Collaboration (DESC)
- MicroBooNE
- MINERvA
- MINOS
- Mu2e
- Muon g-2
- NOvA
- ORKA
- PHENIX
- STAR
- T2K

Preview for ATLAS

Experiments <http://hepsoftware.org/e/atlas>

ATLAS is a particle physics experiment at the Large Hadron Collider at CERN that is searching for new discoveries in the head-on collisions of protons of extraordinarily high energy. ATLAS is learning about the basic forces that have shaped our Universe since the beginning of time and that will determine its fate. Among the possible unknowns are extra dimensions of space, unification of fundamental forces, and evidence for dark matter candidates in the Universe. Following the discovery of the Higgs boson, further data will allow in-depth investigation of the boson's properties and thereby of the origin of mass.

- Contact [Eric Lancon, Computing Coordinator](#)
- Contact [Simone Campana, Deputy Computing Coordinator](#)
- Collaboration website <http://atlas.web.cern.ch/Atlas/Collaboration/>
- ATLAS public web <http://atlas.ch/>
- ATLAS Software Technical Meeting (open beyond ATLAS) 2015-11-09
- @ATLASexperiment
- YouTube

Tags

News updated recent

Science fields

LHC, collider physics ATLAS science field LHC, collider physics

Associated with

- BNL RHIC ATLAS Computing Facility (RACF) ATLAS Tier 1 Center
- CERN ATLAS is located at CERN's Large Hadron Collider (LHC)
- Università degli Studi di Milano

ATLAS uses

- AthenaHive AthenaHive is ATLAS' multithreaded offline framework
- FAX FAX is the basis for ATLAS' xroot based federated data store
- GaudiHive GaudiHive is basis for ATLAS multithreaded framework AthenaHive
- Geant4 Geant4 is the basis for the ATLAS detector simulation
- Gooda ATLAS uses Gooda
- HepMC ATLAS uses HepMC
- Jenkins Various ATLAS software projects use Jenkins for continuous integration, and it is being considered for general adoption
- PanDA PanDA is the basis for ATLAS' distributed analysis and production workload management
- ROOT ATLAS uses ROOT
- XRootD XRootD is the basis for ATLAS remote data access and federated storage

Reporting they are used by ATLAS

- Apache Flume Apache Flume used by ATLAS

ATLAS

Nickname:

The nickname establishes a simple url of form <http://hepsoftware.org/e/atlas> to reference this page. The nickname must be unique.

Type:

Subtype:

Description:

Content format:

Content:

ATLAS is a particle physics experiment at the Large Hadron Collider at CERN that is searching for new discoveries in the head-on collisions of protons of extraordinarily high energy. ATLAS is learning about the basic forces that have shaped our Universe since the beginning of time and that will determine its fate. Among the possible unknowns are extra dimensions of space, unification of fundamental forces, and evidence for dark matter candidates in the Universe. Following the discovery of the Higgs boson, further data will allow in-depth investigation of the boson's properties and thereby of the origin of mass.

Attributes

- Url Collaboration website <http://atlas.web.cern.ch/Atlas/Collaboration/>
- Contact Contact [Eric Lancon, Computing Coordinator](#)
- Contact Contact [Simone Campana, Deputy Computing Coordinator](#)
- Url ATLAS public web <http://atlas.ch/>
- Twitter @ATLASexperiment <https://twitter.com/ATLASexperiment>
- Other YouTube <https://www.youtube.com/user/TheATLASexperiment>
- Event ATLAS Software Technical Meeting (open beyond ATLAS) 2015-11-09 <https://indico.cern.ch/event/395887/other-view?view=standard>

- Add attribute of type
- Website
 - Contact
 - Wiki
 - Documentation
 - Url
 - Reference
 - Presentation
 - Repository
 - Github
 - Git
 - Bitbucket
 - Issue tracker
 - Download
 - License
 - Forum
 - Blog
 - Event
 - Event series
 - Training
 - WikiToLearn
 - Wikipedia
 - Twitter
 - Reddit
 - Content credits
 - Other

Relations

HSF Technical Notes

- Technical Notes can be proposals, ideas, whatever people want to add
- First TN with the TN policy has been published
- Some more in preparation:
 - Licence Guidelines, Naming conventions, packaging tools landscape, ...
- Repository and version control in [GitHub](#)

TN Number	Title	Authors	Download
HSF-TN-2015-01	HSF Technical Notes policy	A. McNab	PDF GitHub

Drafts in the acceptance process:

Draft TN Reference	Title	Authors	Download
HSF-TN-2015-LIC	(Draft) Software Licence Agreements HSF Policy Guidelines	J. Harvey et al.	GitHub
HSF-TN-2015-NAM	(Draft) HSF Platform Naming Conventions - A Proposal	B. Hegner	GitHub
HSF-TN-2015-PKG	(Draft) HSF Packaging Working Group Report	B. Hegner, L. Sexton-Kennedy	GitHub

Training

- People having knowledge are rare
- People having time to actively share their knowledge even more rare
 - Not a problem of motivation!
- So how to make best use of what is there?
 - First of all - make it visible!
 - Lower the bar for collaborative editing and re-use
- Visibility of training material
 - The Knowledge Base is the place for advertising things
- Easier collaborative editing
 - Put your material under a Creative Commons license
 - HSF invested into WikiToLearn

join the [hep-sf-training-wg](#)

- **WikiToLearn** is a wiki-based platform tailored at training and teaching
- Initiated in the context of italian universities
 - Basic idea was that students can improve and extend the material of their professors, while still being quality-controlled
- HSF jumped onto that to see whether we can take advantage of it
 - Started adding material to this site
- Now investing in providing interactive tutorials
 - think of the combination of jupyter style notebooks and a privately owned sandbox - start tutorial now, resume later (this even triggered a new collaboration w/ the ROOT team)
- This is only the shell, content has to come by the community
 - **Is there material our group can contribute?**



Software Packaging

- Topics
 - package building, deployment, runtime environment, new technologies like Dockers, cmake best practices
- Organized a series of discussions/presentations on packaging and build tools (7 meetings)
 - Current practices inside and outside HEP
 - Document to summarize findings being prepared
- Trying a hands-on approach to increase share of actual code even if existing experiments and projects locked-in to a certain packaging solution
 - Common “build recipes” protocol

join the [hep-sf-packaging-wg](#)

Build and Packaging Software Review

Looked at many tools, in particular

- worch, cmsBuild, aliBuild, LCGCMake, SciSoft, contractor (HEP)
- homebrew, Nix, conda (most promising non-HEP)

and compared them to our requirements.

Main problems in HEP software

- reinvention of the square wheel
- non share even within the community

	Multi-Rel	Multi-BuildVar	MultiShell-RTE	Relocation
cmsBuild	+	+	+	+
Contractor	+	+	?	?
Homebrew	-	-	NA	-
LCGCMake	+	+	+	+
Nix	+	-	+	-
SciSoft	+	+	+	+
Worch	+	+	+	+

Main problems in non-HEP software:

- non HEP-tools prefer rolling releases / care less about reproducibility
- little support for multi-environment setups

Conda seems one of the better non-HEP candidates

- One of their developers participated in our last meeting
- Further follow up planned

Evolving document available at <https://github.com/HEP-SF/documents/tree/master/HSF-TN/draft-2015-PKG>

Common 'build recipes'

- Most of our build and packaging work goes into adjusting build instructions to newer compilers and options
 - all encoded in scriptlets
- Though LHC experiments have a common discussion forum (Librarians and Integrators Meeting), the share of work in this area is surprisingly now
- Idea is to define a common protocol to share which consists of
 - metadata (in yaml)
 - build instructions (as script)
- **Proof-of-principle done** with some aliBuild / LCGCMake examples

```
package: libunwind
version: 1.1-%(short_hash)s
source: https://github.com/igprof/libunwind
tag: master
requires:
- libatomic_ops
```

metadata

```
#!/bin/sh
(cd $SOURCEDIR && autoreconf -i)
$SOURCEDIR/configure \
CPPFLAGS="-I$LIBATOMIC_ROOT/include" \
CFLAGS="-g -O3" \
--prefix=$INSTALLROOT \
--disable-block-signals
make ${JOBS+-j $JOBS}
make install
```

instructions

Software Projects

- The essence of the Foundation are the **Software Projects** under its umbrella
 - HSF does not enforce any particular software process, project management or methodology, however packages should conform to some standards to facilitate integration
- Defined preliminary [Project Guidelines](#)
 - Project name, public repository, web site, issue tracker, version naming, mandatory documentation, best practices,...

Software Project Templates

- The idea is to develop a **project template** implementing these guidelines and best practices
- To support small projects that do not have a collaboration environment available
- To serve as example for shared projects across collaborations
 - reducing impedance mismatch
 - helping developers to focus on important things
- Prototype template available on [GitHub](#) covering doxygen, unit tests, CPack, CTest, license files, ...
- Input to this is very welcome!

Shared Software Projects

- Shared software projects can go from almost trivial to really sizeable
- Example for a trivial one:
 - [cmaketools](#) as collection of FindXXX.cmake macros
- A not so trivial one:
 - Gaudi
- Just starting ones:
 - Common Simulation Interface (LHCb / FCC)
 - Next-generation conditions data project (ATLAS/CMS effort)

Conclusions

- We have made quite some progress in some areas but at a slower pace than anticipated
- Areas for which we would like to have more help are:
 - Engagement of some more software projects and development of a project template
 - Fill the knowledge base!
- Please join and contribute to any of the working groups, the startup team, discussion forums
 - subscribe to the fora to follow progress and contribute