SEVENTH FRAMEWORK PROGRAMME

Research Infrastructure

FP7-INFRASTRUCTURES-2010-2 – INFRA-2010-1.2.3:

Virtual Research Communities

Combination of Collaborative Project and Coordination and Support Actions (CPCSA)

LinkSCEEM-2

Linking Scientific Computing in Europe and the Eastern Mediterranean – Phase 2

Grant Agreement Number: RI-261600

D4.5

Report on adapting needs and emerging trends in user training needs

Final

Version: 1.0
Author(s): Alan O'Cais, JSC
Date: 26/02/2015

Project and Deliverable Information Sheet

LinkSCEEM Project	**Project Ref. №:** RI-261600	
	Project Title: LinkSCEEM-2	
	Project Web Site: http://www.linksceem.eu/ls2/	
	Deliverable ID: < D4.5>	
	Deliverable Nature: <DOC_TYPE: Report>	
	Deliverable Level: PU *	**Contractual Date of Delivery:** 28 / 02 / 2015
		Actual Date of Delivery: 17 / 03 / 2015
	EC Project Officer: Leonardo Flores Anover	

* - The dissemination level are indicated as follows: **PU** – Public, **PP** – Restricted to other participants (including the Commission Services), **RE** – Restricted to a group specified by the consortium (including the Commission Services). **CO** – Confidential, only for members of the consortium (including the Commission Services).

Document Control Sheet

Document	**Title:** **<Report on adapting needs and emerging trends in user training needs>**	
	ID: **<D4.5>**	
	Version: <1.0 >	**Status:** Final
	Available at: http://www.eniac.cyi.ac	
	Software Tool: Microsoft Word 2007	
	File(s): LinkSCEEM-2-D4.5 Report on adapting needs and emerging trends in user training needs.docx	
Authorship	**Written by:**	Alan O'Cais, JSC
	Contributors:	WP 4 participants
	Reviewed by:	
	Approved by:	PMO

Document Status Sheet

Version	Date	Status	Comments
1.0	26/02/2015	Final version	
1.0	03/03/2015	approved	

Document Keywords

Keywords:	LinkSCEEM-2, Computational Science, HPC, e-Infrastructure, Eastern Mediterranean

Table of Contents

Executive summary

The practical implementation of the training program of LinkSCEEM has matured throughout the project lifetime, elaborating on the core concepts of basic and advanced user training that formed part of the original project Description of Work.

The project has worked towards a model of user training where the regional community collaborates in the training effort, resulting in a number of mutually beneficial results including:

- Distribution of effort and knowledge;
- Leveraging of all regional expertise;
- Standardisation of training material;
- Standardisation of computing environments;
- Creation of a support network for the instructors themselves.

The project has adapted to the specific needs of its user communities by considering the fact that the number of developers of HPC applications of production quality in the region are relatively few, and those that do exist are already greatly experienced. Instead, the project has recognised that the vast majority of production usage comes from community-developed applications and has adapted its training program to reflect this reality.

From this point of view, basic training has come to represent the core skills that are required to create and use an efficient workflow on the HPC systems provided by LinkSCEEM. This includes aspects such as:

- Automating tasks using the Unix shell;
- Structured programming;
- Version control;
- Data management;
- Queuing systems;
- HPC architectures.

Advanced training was tuned to be more specific to user community needs with a number of workshops focussed on particular scientific areas, organised in parallel to each other.

There has also been a continuous effort to look towards a sustainable training model, one that creates a community of the HPC support staff in the region. This model includes

- Use of Software Carpentry[1] instructor training courses to provide trainers with basic knowledge of educational psychology and instructional design;
- Collaborative development of training content;
- Creation of an effective platform for remote training, the Supercomputing Training Portal[2].

The project has also recognised the importance of performance analysis when it comes to effective usage of HPC resources. Combining tools such as XALT[3] (a job level tracker) and LWM2[4] (a performance analysis tool), one has a powerful and **passive** ability to analyse the runtime utilisation of compute resources and perform some basic performance analysis of user applications. These

---

[1] http://software-carpentry.org/

[2] http://supercomputing.cyi.ac.cy/

[3] https://github.com/Fahey-McLay/xalt

[4] http://www.vi-hps.org/Tools/LWM2.html

methods can expose many factors that typically remain hidden from the administrators and user support of HPC systems and provide them with a starting point regarding how to engage with the user community to more effectively use the computing resource.

1 Introduction

The original program of the training work package had two basic subdivisions:

- **Basic user training** which was intended to cover a range of fundamental topics needed by a wide spectrum of the computational science community. Such topics include parallel programming models and tools, and introduction to visualization techniques. The training was to be offered via courses, hands on workshops and online tutorials using modules already developed by NCSA and FZJ-JSC. The hands-on workshops were to provide training in both the development and operations phase in topics ranging from discipline-specific strategies to advanced programming techniques required to exploit available computational resources.
- **Advanced user training** programs provide experienced users with education in both general interest and discipline-specific topics. General interest training includes emerging HPC technologies and tools, programming techniques for GPUs and multi-core programming for homogeneous and heterogeneous architectures. One thematic workshop per year per field will be organized at SESAME and other regional institutions. The programs were designed to encourage active trainee participation and promote hands-on activities.

One of the primary tasks of the work package was to adapt this training program to suit the needs and requirements of the user communities themselves. In this report, we look at the development of the training program throughout the lifecycle of the LinkSCEEM project and see how the program was adapted and expanded upon to meet the needs of the user communities that partook in it.

We also look at some of the globally emerging trends in HPC-related training and how they have impacted the approach to user training. We describe how LinkSCEEM attempts to leverage such developments to ensure the sustainability of its training program.

In the final section, we outline the approach that LinkSCEEM takes towards the future of a regional training program.

2 Adapting Needs

The general merits of the initial training program of LinkSCEEM are clear. In practice, the diversity of the user communities that engaged with the project proved to be quite a challenge. Some of the users were accustomed to using some of the largest supercomputers available in the US, while others had never used anything other than a Windows-based desktop.

This diversity forced a rethinking and streamlining of the training program so that the majority of the training effort was focussed on making scientists' lives easier rather than spending time on small technical details that can be easily solved with a quick query to a search engine.

2.1 Basic Training

The dynamism and breadth of HPC technologies and application development poses a challenge when attempting to create comprehensive training content for new or existing user communities. The list of software and hardware technologies that are relevant to such content is as long as you choose it to be because HPC is used by such a wide range of user communities, each with their own characteristics.

In this section, we describe how LinkSCEEM has adapted itself to this reality for the purposes of basic training. The project sought to find, as much as possible, the common denominators among each of the various user communities and provide the basic skills which would be required to understand the capabilities of the HPC resources provided to them, as well as how to use them effectively.

2.1.1 *HPC Roadshow*

The opportunities to engage in person with the potential LinkSCEEM user communities are relatively rare. Many people in this community are intimidated by the fact that they have never used a HPC resource in the past. However, factors such as a lack of experience of Linux, or never having compiled a piece of code before are, in reality, not a significant barrier to using HPC resources. Unfortunately this cannot be realised by someone within making that first leap and actually attempting to use such resources. Removing this hurdle for the true beginner was one of the things that LinkSCEEM has tried to do.

The project realised that the User Meetings organised within the project, where project partners toured the region informing people about LinkSCEEM and the compute resources available within it, were the perfect platform for a short introductory training session. Over time, this session has been crafted to be short and to the point, avoiding technical details as much as possible and getting people to the stage where they have successfully submitted a job to the cluster and received a result (within a 2 hour training window). A number of developments within the project have allowed this to be possible:

- Many of the users have never seen (or heard of) the Linux operating system. Since most HPC systems run on this environment, this in and of itself is a huge barrier. Having to type commands rather than click a mouse is something new for many people. The adoption of the Supercomputing Training Portal within the HPC Roadshow allowed us to sidestep many of these issues:
    - The Portal can open a terminal session within the users browser, therefore it is completely agnostic as to what operating system they are using.

  - Internally the Portal leverages public/private key cryptography, but to the user it presents a simple password interface. This allows us to sidestep the explanation of how such cryptography works and how to use it on different operating systems.
- The examples used within the Roadshow have been prepared in detail. They do not require any modification on the part of the user. They simply require that the user develops some general understanding of how the cluster environment works:
  - Using the *module* environment (typical of HPC systems) to access installed software, such as a compiler
  - Compiling an application
  - Editing a file, creating a job script and submitting a job to the resource manager
  - Querying the queuing system and waiting for a job to finish, understanding that a job is actually executed on a remote resource
  - Reading the output from a job

Success in such examples gives people the confidence needed to consider engaging with the next level of training provided by the project. It also introduces to them to the User Support team of the project in a way that makes that team appear accessible.

2.1.2 *Understanding HPC Architecture*

The General User Meetings of LinkSCEEM have provided the platform for another important aspect of user training, that of having a basic understanding of what a parallel program actually does and how that is related to the computer architecture that the program runs on.

The resources available within LinkSCEEM ultimately consisted of cluster-architectures with GPU accelerators available at the Cyprus Institute. The programming models taught during these User Meetings reflected this and covered topics such as:

- MPI – which highlights the distributed memory nature of the compute resources
- OpenMP – which leverages the multicore architecture typical of modern CPUs
- Accelerating coprocessors – which introduce an additional layer of complexity but are also computationally the most powerful component of a compute node (when used correctly)

For the specific case of accelerators, the technologies to leverage such hardware were developing throughout the lifetime of the project, allowing us to move from the relative complexity of CUDA for the beginner to the more familiar environment of OpenMP 4.0 (which supported accelerators) by the end of the project.

It is an important point to note that many of the users of the LinkSCEEM resources are unlikely to ever engage in code development themselves. This is clear when you analyse the applications for Production Resources received by the project, where a majority of applicants are typically end-users of a community-driven application that is already enabled for HPC architectures (such as GROMACS or NAMD). That does not mean that such training events are of no value to them, indeed the reality is quite the opposite. An introduction to these topics provides them with a basic understanding of how compute resources are being used by an application, and this knowledge allows them to more effectively understand how to prepare their use of a community code to utilise resources more effectively:

- How to run MPI-enabled applications
- Scaling across multiple nodes and what to expect from it
- Minimising communication overhead by using an OpenMP-enabled mode when available
- Enabling GPU support within an application and what to expect as a result

- Understanding the language used within an applications documentation (threads, cores, accelerators, shared memory, distributed memory)

2.1.3 *Software Carpentry*

Computing is now an integral part of every aspect of science, but most scientists are never taught how to build, use, validate, and share software well. As a result, many spend hours or days doing things badly that could be done well in just a few minutes. During the project lifetime, we became aware of the Software Carpentry Foundation[5] which is a volunteer organization whose goal is to make scientists more productive, and their work more reliable, by teaching them basic computing skills. It runs short, intensive workshops that cover core topics in program design, version control, testing, and task automation so that scientists can spend less time wrestling with software and more time doing useful research. Some topics which are covered include:

- Automating tasks using the Unix shell;
- Structured programming in Python, R, or MATLAB;
- Version control using Git or Mercurial; and
- Data management using SQL.

The Software Carpentry group is not focussed on HPC, but it touches on many of the factors that impact the typical working day of a scientist using HPC resources. The fact that the organisation is a collaborative volunteer effort with well documented and maintained training content in its core curriculum makes integrating its concepts into basic user training relatively straightforward and is something that we have done in particular in the last year of the project, where we have held 2 Software Carpentry training events as part of larger training programs.

Another valuable resource Software Carpentry can provide is ***instructor training***, which is free and conducted online. It teaches scientists how to teach other scientists how to program. In this training course, they cover the basics of educational psychology and instructional design, and look at how to use these topics in both intensive workshops and regular classes. The training takes 2-4 hours/week for 12-14 weeks. The concept is that you don't need to be an expert programmer to take part: in fact, they prefer instructors who are only two steps ahead of their learners and can still remember what it's like to find programming difficult and confusing.

One of the LinkSCEEM team has already qualified as a Software Carpentry instructor and has found it to be an extremely valuable exercise. In future we hope to leverage this instructional training course for all user support staff.

2.2 Advanced Training

The approach to advanced training has developed quite significantly over the lifetime of the project. The project has coordinated a number of training events with the Partnership for Advanced Computing in Europe[6] (PRACE). This has given the project access to some of the most experienced instructors in Europe, and indeed to some of the most advanced topics in HPC.

---

[5] http://software-carpentry.org/

[6] http://www.prace-ri.eu/

The project has also taken a very pro-active position towards advancing the user communities within specific domains, in recognition of the fact that many of the typical users of Production class resources are users, and not developers, of HPC applications.

Performance analysis is also a critical skillset for any user interested in optimising an application that they develop themselves, or even a community-developed application that they use.

Finally, the project has recognised the role that administrators play on an ongoing basis to the quality and impact of the computational resources available to research scientists; and the ability of the scientists to use them effectively.

In this Section we discuss all of these developments.

2.2.1 *Joint PRACE Training Events*

The initial advanced training event of LinkSCEEM was conducted in coordination with PRACE in January 2011. Its program was very broad, covering a host of topics including:

- Advanced MPI;
- Parallel application design;
- New programming languages and extensions: PGAS (partitioned global address space) languages and OpenCL (for accelerators);
- Performance analysis;
- Visualisation;
- Debugging.

The project went on to hold a number of joint events with PRACE over the project lifetime, most recently the final Cross-Disciplinary Advanced Workshop in Israel in February 2014. The topics covered remained at a very high level with continued focus on aspects that are most likely to influence users within LinkSCEEM resources.

2.2.2 *Community-developed Applications*

Preparatory access for LinkSCEEM HPC resources opened in March 2011 with the first Production Call for Proposals opening in December 2011. From an early stage, it was clear that there was not a large development community in the region and that, for Production usage in particular, most applicants were end users of community-developed application codes.

For this reason, an initiative was taken to focus the Advanced Workshop in February 2012 on these application codes themselves, with the idea of preparing the users for the Production Call running at that time. Parallel training sessions in selected areas were targeted to the participant's interests:

- Computational fluid dynamics (featured software: OpenFOAM);
- Graph partitioning (featured software: Metis/ParMetis);
- Molecular Dynamics (featured software: GROMACS);
- Computational Biology & Bioinformatics (featured software: MrBayes & RAxML).

This program proved to be highly popular with the applicants and a similar approach was adopted during the 3rd Cross-Sectional workshop in Cairo in November 2012 and again in at the General User Meeting in June 2013 covering the additional topics:

- Structural and Computational Biophysics (featured software: NAMD2);
- Real-Time Density Functional Theory (DFT) based Simulation (featured software: Octopus);
- Scientific Simulation and Analysis (featured software: Geant4 & Root);

- HPC for Bioinformatics;
- Molecular Dynamics Simulation tools (LAMMPS);
- Materials Science (Quantum-Espresso).

GPU-enabled community codes

The project recognized that the level of GPU usage within the project was neither diverse enough across subjects nor widespread enough through the community of users. As a result, a distinct effort was made to increase the utilization of GPUs among users through the organization of a GPU workshop targeting users of applications that were already GPU-enabled. 9 attendees were invited based on the applications which they utilized on the LinkSCEEM HPC resources. The workshop was held from the 10th to the 12th of December 2012.

The applications that were addressed were:

Gromacs - http://www.gromacs.org/Documentation/Installation_Instructions/GPUs
NAMD - http://www.ks.uiuc.edu/Research/namd/cvs/notes.html
DL_POLY_4 - http://www.ccp5.ac.uk/DL_POLY/MANUALS/USRMAN4.pdf (section 1.6)
AMBER - http://ambermd.org/gpus/
LAMMPS - http://lammps.sandia.gov/doc/Section_accelerate.html

There were 5 different Production projects represented at the workshop and many of them had overlapping applications (NAMD and GROMACS in particular were common to many). Overall, attendees worked with their own projects but this overlap proved very helpful to some.

2.2.3 *Performance Analysis*

The project has consistently tried to educate people on the basics of Performance Analysis, whether that be at a detailed level for code development (using tools such as Scalasca[7]) or at a more basic level simply to understand if resources are being used effectively (with tools such as LWM2[8]).

Sessions on performance analysis have appeared in almost all of the advanced schools held by the project and also in some of basic training programs.

2.2.4 *Administrator Training*

In the final Advanced Workshop of the project, held in January 2015, focus was given to the regional system administrators who manage the resources that scientists use. System administrators are a crucial component in any system since they are the ones that provide the basic software used by the scientists and in many cases are the ones installing community-based codes used by the majority of users.

The subjects delivered include SLURM (Simple Linux Utility for Resource Management), an overview of HPC monitoring systems and general parallel file systems, LinkTest, EasyBuild, JUBE and Big Data in HPC. Trainers for these sessions came from Bull (France), Jülich Supercomputing Centre (Germany) and NCSA (USA).

[7] http://www.scalasca.org/

[8] http://www.vi-hps.org/Tools/LWM2.html

3 Emerging Trends

The HPC community it growing quickly as its applications and capabilities become more accessible to researchers. In the specific case of the Eastern Mediterranean, it has significant added value since the “laboratory” (that is the HPC) resource is a tool that can be uniformly accessible across the region by means of the internet. This is very significant since it allows all researchers to have access to facilities of an international standard, even when research infrastructure within a country may not typically be of this standard.

With the emergence of powerful community-based applications in many fields, whose use does not necessarily require hard-won HPC skills, the barrier to effective utilisation of HPC resources has never been lower. Nevertheless, computational scientists of the current era still require a general computational skillset to maximise the efficiency of their workflow in these environments. This is an issue that is certainly not particular to the region and there are significant efforts worldwide to ensure that scientists are both aware of and in a position to master these skills.

In this section we look at some of the emerging trends in this area:

- direct provision of targeted training to scientists by ‘local’ trainers;
- self-learning via online resources;
- automated tools that facilitate extracting information on how computational scientists use HPC systems and thereby advise them on how to optimise their usage.

3.1 Localised Training Expertise

It is easy to be intimidated by high performance computing when you first approach the subject. For many, the move from mouse-clicks to the command line is very challenging and concepts such as distributed memory are difficult to grasp. This first hurdle is something that can quite easily be overcome if you have direct assistance when you really need it.

With the HPC Roadshow of LinkSCEEM, we spent significant effort creating standardised training content to introduce people to an HPC system. As far as possible, we removed technical details and tried to reduce tutorials to only the necessary core concepts.

Once we had training content we were satisfied with, those who created that content made a video of themselves delivering that content. The content itself together with the video of its presentation was then shared with those at BA offering Roadshow events in Egypt and SESAME who offered Roadshow events in Jordan. The local trainers were then able to practice the content privately and deliver it themselves at their own events. This concept of **standardising the content** and **localising the trainers** has proven to very powerful, it has allowed LinkSCEEM to greatly increase the number of User Meetings in each country since the level of effort involved is now relatively low. It has also exposed the **local community** to the **local support** in their area, increasing their perception of the accessibility of HPC.

3.1.1 *Software Carpentry*

This concept of collaborating on training content and localising the trainers is not unique to LinkSCEEM and, as we discovered, is something that has been going on since 1998 within the

Software Carpentry Foundation[9]. As described previously, the goal of Software Carpentry is to teach computational scientists the basic computing skills that they need to do their work efficiently.

On top of providing collaboratively generated content on a number of topics that are highly relevant to HPC users, the organisation also offers to remotely train instructors so that they can deliver that material in an effective manner. Given that many of the prospective regional trainers we envision would have little or no teaching experience, the ability to provide them (remotely and at zero cost) with proven core concepts in educational psychology and instructional design to effectively deliver material is an incredible prospect.

3.1.2 *Our Vision*

Taking all of these developments into account, our vision is to create standardised content in the "Basic Training" topics that were described in Section 2.1. This content would then be delivered regionally by local instructors who have gone through the Instructor Training provided by the Software Carpentry Foundation.

These instructors would then form a regional support network for computational scientists using the available HPC resources. The network would also form a collaborative framework for the instructors themselves, providing them with a set of peers that can help them solve problems coming from their local research community.

3.2 Online Training

In the previous section we discussed how basic training could be sustainably provided within the region, but what of the case of material that goes beyond the introductory? Financially speaking, delivering this type of material is an expensive prospect, high quality trainers with the necessary knowledge have to be flown in to particular location and the trainees also have to reach that location. Many regional institutions do not have travel budgets for such activities so attendance to any such event is likely to be highly localised.

From relatively early within the project, it was realised that online training is by necessity an important component of any training program. For basic, but in particular advanced, training we must provide people with adequate means to educate themselves.

3.2.1 *Supercomputing Training Portal*

For this reason, the development of the Supercomputing Training Portal[10] became an important component of the project. There were a number of important aspects to take into account in the design of the portal:

- The quality of regional connectivity fluctuates greatly, any solution must not have high bandwidth requirements;
- Without an easily accessible platform to attempt practical content, the effectiveness of the portal would necessarily be limited;
- There is already a huge amount of high quality training content available throughout the web, within, for example, MOOCs such as Coursera[11] and organisations such as PRACE[12].

[9] http://software-carpentry.org/

[10] http://supercomputing.cyi.ac.cy/

The design of the portal separates the collection and cataloguing of training content from the presentation of that content to the end user. In addition, it provides a means of performing hands-on exercises through a web-integrated terminal emulator that connects directly to a true HPC resource.

Training content within the portal is classified as:

- Web-based material, where the content is a (series of) web-page(s) or a web-application;
- Captured material, where a lecture (or screen) capture has taken place;
- Documents, such as a PDF of a presentation or perhaps a detailed training document.

Within the Training Portal, such content is collected, categorized and given a set of meta-data that can be used to manipulate the content when presenting it to the end user.

Adopting common tools and implementation approaches when it comes to practical examples could potentially lead to further fruitful training collaborations. For example, a dedicated educational cluster with accelerators is provided by the Cyprus Institute. The cluster uses EasyBuild[13], an HPC software build and installation framework which can be leveraged to provide a homogenized execution environment for practical material.

3.2.2 *Our vision*

The localised instructors of section 3.1 would advertise the Portal to those attending live training events as a means to continue and advance their training. The trainers themselves would collect further relevant material from across the web and integrate it into the portal. Instructors can then also use the contents of the portal to create tailored training course for particular communities, something that can be easily facilitated by the capabilities of the proposed content repository Medici[14].

Regional advanced training events will still be held, but they will also be captured using methods developed within the LinkSCEEM project. Captured material will then be added to Medici and processed there. Medici has the ability to also hold practical material side-by-side with the captured content allowing people to easily access this essential component of any training event.

3.3 Passive Analysis of Resource Utilisation

The performance of any application on HPC resources is highly volatile and dependent on factors such as system architecture, application parameters, I/O bandwidth, MPI implementation, process pinning, etc. It is clear therefore that when one is considering the effective use of HPC resources, the issue of application performance analysis must be addressed. Unfortunately, the reality is that many users never systematically address the performance of their code.

Luckily, a number of tools are being developed that passively analyse system utilisation and application performance, adding this information to a database that can be analysed on both system and user levels. We review two such tools in the following sections.

---

[11] https://www.coursera.org/course/scicomp for example

[12] http://www.training.prace-ri.eu/tutorials/index.html for example

[13] https://github.com/hpcugent/easybuild

[14] https://opensource.ncsa.illinois.edu/projects/MMDB

3.3.1 *XALT*

As described elsewhere, many computing centres provide the computing environment, and even the applications used by the majority of users, through *environment modules*. Tools such as EasyBuild[15] allow centres to collaborate on the HPC software build and installation procedure for these tools, providing a standardised and potentially vast module environment.

Inevitably, there are application and software developments that require the upgrading of compilers or applications, or bug-fix reinstallations. For such reasons, most computing centres would like to know

- How many users and projects use a particular library or executable;
- If a library they maintain is used (and how often);
- If centre provided packages are used more or less than user-installed packages;
- After the fact identify users and code that used a buggy library;
- Provide information on how an executable was built (provenance data);
- Catch compile-time/run-time differences in a user's environment;
- Identify applications that are using deprecated libraries and identify outdated binaries.

XALT[16] is a job level tracker and census taker. It records what executables and libraries users run on a compute resource. Those libraries and executables can be tracked back to the environment modules that they came from. The recorded data can be used to know what programs and libraries are (or are not) used. This not only makes managing a system easier, but also provides the centre with information that can potentially be used to assist the users in improving the performance of their applications.

3.3.2 *LWM2*

The lightweight monitoring module LWM2[17] is a low overhead profiler. It can profile applications without any modification by a user and can operate in a passive manner, sending the extracted information to a central database.

The lightweight measurement module uses a hybrid approach to profile an application. It samples the profiled application at regular intervals to keep track of application activity. To keep the overhead low, LWM2 avoids stack unwinding at each application sample. Instead, it utilizes direct instrumentation to earmark regions of interest in an application. When an application is sampled, the earmarks are checked to identify the region of application execution. As a result, LWM2 is able to profile an application with reasonable knowledge of application activity while maintaining low overhead. This hybrid approach also allows LWM2 to keep track of the time spent by an application in different regions of execution without directly measuring the time in these regions. The hybrid profiling approach is also used to collect additional data of interest for some specific application activities. This includes:

- the MPI communication calls and the amount of data transfer;
- the POSIX file I/O calls and associated data transfers;
- the performance of multithreading in an application;

[15] https://github.com/hpcugent/easybuild

[16] https://github.com/Fahey-McLay/xalt

[17] http://www.vi-hps.org/Tools/LWM2.html

- sequential performance information by profiling the system hardware counters;
- using the CUPTI interface to profile CUDA applications.

3.3.3 *Our vision*

Combining XALT and LWM2, one has a powerful and **passive** ability to analyse the runtime utilisation of compute resources and perform some basic performance analysis of the user's applications. These methods can expose many factors that typically remain hidden from the administrators and user support of HPC systems and provide them with a starting point with how to engage with the user community to more effectively use the computing resource.