

<center><span style="font-family: verdana; font-size: 20px; color: #275c86;">Montenegro, Budva, Becici, 28 september - 02 october 2015</span></center>



Contribution ID: 97

Type: **not specified**

## The evolution of the WLCG Grid

*Monday, 28 September 2015 12:10 (30 minutes)*

The Worldwide LHC Computing Grid (WLCG) has been in production for more than 10 years supporting the preparations for, and then the first run of the LHC. It has shown itself to be one of the pillars of the infrastructure necessary to enable the rapid production of physics results from the LHC, and has been in constant use at a very high load since its first introduction. However, even from the first months of real data flowing in 2010, the computing models and the WLCG infrastructure itself have been evolving to adapt to the realities of real data, and the real use cases of the experiments. In particular, the data management services have responded to the significant capabilities of the global network available to LHC, far above that anticipated, and the requirement to optimise data placement and movement. Concepts such as global data federations and intelligent data placement and caching have been introduced. In recent years, virtualisation and cloud technologies have become more and more important and are now an important piece of the WLCG technology. Since the experiments and WLCG itself receive offers of computing not just from the pledged resources, but also in the form of opportunistic resources in private and public clouds, in HPC machines, and various other sources such as volunteer computing, the foreseen evolution of WLCG must be to make use of this pool of opportunity, and to not restrict itself to “grid” or “cloud”, but to adapt and easily incorporate heterogeneous resources as they are made available. This talk will summarise the experience of Run 1, and how the WLCG is anticipated to evolve during Run 2 and in preparing for the LHC and detector upgrades.

**Primary author:** BIRD, Ian (CERN)

**Presenter:** BIRD, Ian (CERN)