Title	Promoting platform interoperability with portable bcbio workflows
Authors	*Brad Chapman*, Rory Kirchner, Lorena Pantano, Peter Amstutz, Alexander Zaranek, Shannan Ho Sui, Oliver Hofmann
Affiliations	Harvard Chan School Bioinformatics Core (http://bioinformatics.sph.harvard.edu/), Curoverse (https://curoverse.com/), Wolfson Wohl Cancer Research Centre (http://www.gla.ac.uk/researchinstitutes/cancersciences/ics/facilities/wwcrc/)
Contact	bchapman@hsph.harvard.edu
Availability	https://github.com/chapmanb/bcbio-nextgen
License	MIT

Running multi-step analyses requires coordinating biological software and data across a wide variety of heterogeneous computational resources. The complexity of building a parallel workflow representation is a barrier to allowing open source platforms to work together. We've actively developed bcbio (https://github.com/chapmanb/bcbio-nextgen) for the past six years as a open, community built approach to creating variant calling, RNA-seq and small RNA analyses. We'll describe how bcbio runs interoperably within multiple analysis platforms.

We re-engineered bcbio's internal workflow representation to use the Common Workflow Language (CWL: http://www.commonwl.org/). bcbio previously used a parallelization framework build on IPython parallel (https://ipyparallel.readthedocs.org) that runs on both local compute infrastructure (https://bcbio-nextgen.readthedocs.org/en/latest/contents/parallel.html) and on cloud resources (https://bcbio-nextgen.readthedocs.org/en/latest/contents/cloud.html). Having this custom approach meant that we could not easily deploy bcbio on systems like Galaxy (https://galaxyproject.org/) due to different approaches to running compute jobs. This incompatibility results in duplication of effort as bcbio develops and tests system specific parallel code, while the Galaxy community needs to re-implement validated and tested analyses available in bcbio.

By using the community standard CWL for describing workflows, bcbio now runs on Amazon Web Services, Microsoft Azure and Google Compute Engine using Arvados (https://arvados.org/) on clusters using Toil (https://github.com/BD2KGenomics/toil) and locally using cwltool (https://github.com/common-workflow-language/cwltool). Users can choose an infrastructure matching their requirements by building CWL directly from existing bcbio sample description files (https://bcbio-nextgen.readthedocs.org/en/latest/contents/cwl.html). This enables future use of bcbio analysis methods in projects like Galaxy that are actively implementing CWL support.

We'll discuss the challenges of migrating to CWL versus the benefits of being able to integrate within multiple platforms. bcbio is now a better architected, more portable set of validated tools and workflows to help scientists answer biological questions. We focus on developing analysis methods and validations while CWL supporting platforms focus on integration, parallelization and scaling. We hope to promote the continued exploration of ways to re-use and cooperate more effectively as an open source community.