Orchestration

The orchestration software, Orca, allows users to run application code at a larger scale than their own test machine. It coordinates software setup, execution, monitoring and shutdown of the application across multiple machines. Orca has a plug-in architecture, which is used to change how which type of job execution software is used to run jobs. The only currently supported job execution software is HTCondor, but we've use Orca to coordinate job execution natively in the past.

Setting Orca up can be somewhat complex because of the number of configuration files that need to be set up, and because of differences in execution platforms and different user accounts.

In order to make this easier, a configuration writing and execution utility, runOrca.py, was written to be used with Orca. The runOrca.py command takes the specified target platform and fills in the necessary information into configuration files that users would ordinarily have to write themselves if they were using the ctrl_orca package. This allows us to have a single interface to execute across multiple platforms, without having to worry about configuration specifics (home directory location, remote account names, scratch space, etc) for each platform.

The orchestration software is split across multiple packages. These packages are:

ctrl_orca – The orchestration engine, "Orca".

ctrl_execute – configuration writing and orchestration execution

ctrl_platform_* - templates used to write configuration files used by Orca

ctrl_stats – HTCondor specific statistics reporting.