Flexible Cloud Computing through *Swift Coasters*

Ketan Maheshwari, Justin M Wozniak, Mihael Hategan, Allan Espinosa, Daniel Katz, and Michael Wilde
Computation Institute, University of Chicago and Argonne National Laboratory
http://www.ci.uchicago.edu/swift
swift-user@ci.uchicago.edu

Swift is a parallel scripting language for scientific applications

Composes applications linked by files

In use by a large and growing user community

Biochemistry, neuroscience, proteomics, economics, climate research

Easy to write – a simple, high-level, C-like functional language

Small SwiftScripts can do large-scale work

Easy to run: complete Cloud client in one Java application

Unpack and run – a self-contained client for Grids and Clouds

Coasters Enable Flexible Cloud Computing

Coast through clouds from a single submit host via Flexible runtime interface – supports schedulers, Globus, and SSH to Open Science Grid, Bionimbus and Magellan Clouds

Fast and highly parallel

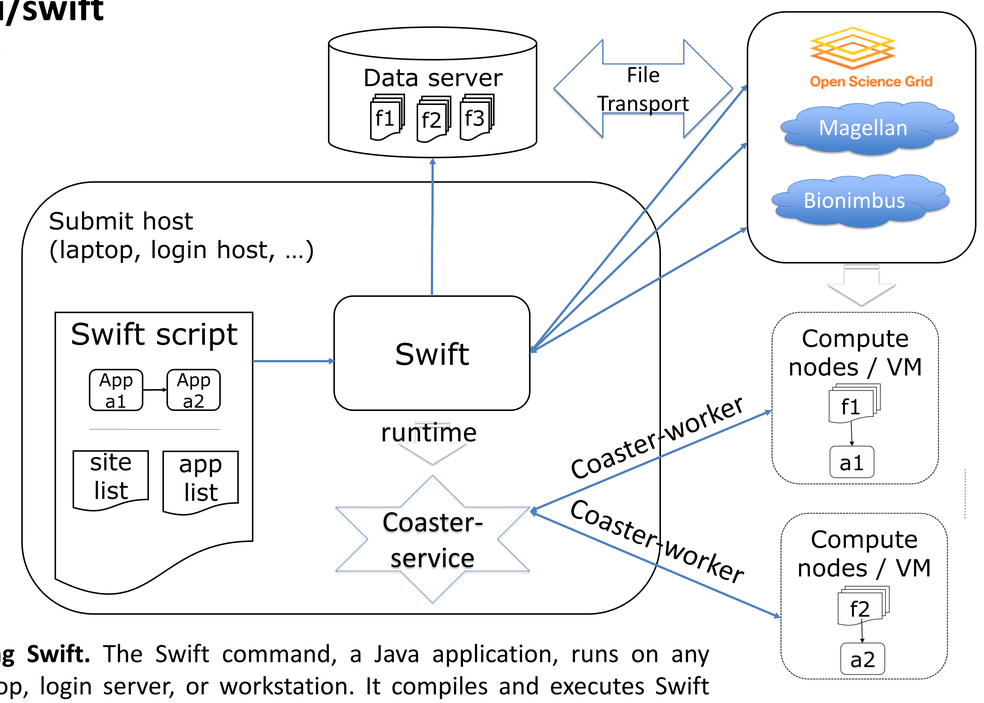
Easily scales to 60,000 tasks on Bionimbus cloud system

Collective data management research is optimizing IO traffic

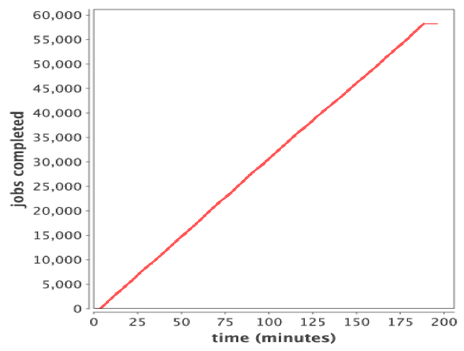
Problem: *How to compose large protein docking workflows, from serial or parallel application programs, to run flexibly on a variety of platforms from a single host?*

Solution: *Here is part of a Swift script for the modftdock application. For 600 proteins, n=100 moves, this script executes 100x600=60,000 docks – and runs on Cloud, Grid, and HPC resources– via Swift Coasters:*

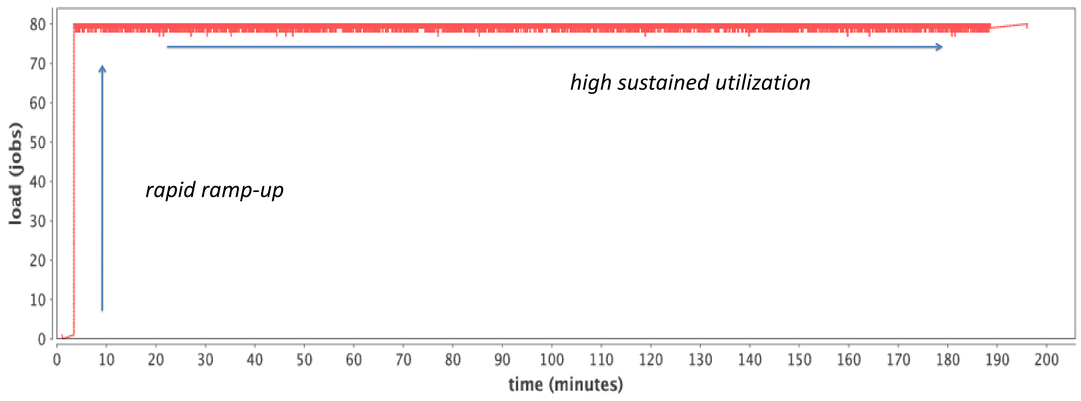
```
foreach str_root in str_roots {
    // break docking jobs + do'em in parallel
    foreach mod_index in [0:n-1] {
        data_files[ mod_index ] =
                do_one_dock(str_root, str_modulo,
                            file_static, file_mobile);
    }
}
```



Using Swift. The Swift command, a Java application, runs on any laptop, login server, or workstation. It compiles and executes Swift scripts, coordinates remote data transfers, and executes applications on local and distributed parallel resources.



modFTDock driven by Swift. "docks" protein to a tRNA molecule with a spatial resolution that controls the granularity of its internal "moves" (in Angstroms). Swift computes the Cartesian product of protein 3D structures against tuples of geometric operators describing a docking conformation by rigid-body translations and rotations against a single tRNA 3D structure. Computation for ~60,000 jobs (left) with a sustained load of ~80 jobs (bottom) on Bionimbus Cloud VMs are shown. *Credit: Marc Parisien, UChicago.*



This research is supported in part by NSF grants OCI-721939 (SDCI) , OCI-0944332 (EAGER), and PHY-636265 (I2U2); NIH grants DC08638 and DA024304; and the U.S. Department of Energy under contract DE-AC02-06CH11357