The HDF5 library is available for AMPI at
https://charm.cs.illinois.edu/gerrit/#/admin/projects/hdf5-ampi

This bug tracks several of the issues that are still needed for complete support.

- Test if applications work with the shared library (currently, only the static hdf5 library is built)
  - This is currently blocked by the lack of a shared-library ROMIO
- SMP mode (seems to work 07/16)
- Virtualization (seems to work 07/16)
- Migration
- Some spurious crashes/segfaults at hdf5 library termination
- Test other architectures than linux/netlrlts. Currently works with:
  - netlrlts-linux-x86_64
  - netlrlts-linux-x86_64 smp
  - multicore-linux-x86_64

For migration, the main concern is migrating with open files: so far we've told users to explicitly close and re-open files before and after migration (or if doing serial I/O, make that rank non-migratable), but we could potentially do that for them in our ROMIO and HDF5 distributions.

What's the status of updating ROMIO to get shared library support?

I'd like to know to know if it is building on AMPI yet, or if it requires any MPI-2 or MPI-3 features we don't have implemented yet, so that I can prioritize them.

Having the ROMIO update on a branch would be nice.
I have a patch to update romio to 1.2.6 (shipped with last version of mpich1) that compiles successfully with the current AMPI and passes all of the romio test suite. More advanced features not currently supported by AMPI (such as generalized requests) are still optional in 1.2.6.

Getting it to actually build a shared library is not so easy though, the current Makefile generates some weird libtool archive that I haven't been able to convert to an .so yet, which is why haven't submitted the patch to gerrit yet.

- Target version changed from 6.8.1 to 6.9.0

HDF5 serial tests working (all 62):

- testhdf5, cache, cache_api, cache_image, cache_tagging, lheap, ohdr, stab, gheap, evict_on_close, farray, earray, btree2, fheap, pool, accum, hyperslab, istore, bittests, dt_arith, page_buffer, dtypes, dsets, ctmpl_dset, filter_fail, extend, external, etc, objcopy, links, unlink, twriteorder, big, mtimes, fillval, mount, flush1, flush2, app_ref, enum, set_extent, tsafe, enc_dec_plist, enc_dec_plist_cross_platform, getname, vfd, ndtypes, dangle, dtransform, reserved, cross_read, freespace, mf, vds, file_image, unregister, cache_logging, cork, swmr, testerror.sh, testlibinfo.sh, testcheck_version.sh

NB: testcheck_version.sh shows some discrepancies in the exit codes returned on failure, but this is not significant for application execution (and can't be fixed for now).

HDF5 parallel tests working ():

HDF5 parallel tests NOT working ():