

api example: saga-file copy

[tools/clutils/file/copy_impl.cpp]

```
saga::filesystem::file file (saga::url(sourceURL), saga::filesystem::Read);  
file.copy(saga::url(targetURL));
```

api example: saga-file list dir

[tools/clutils/file/list_dir_impl.cpp]

```
saga::filesystem::directory dir(directoryURL);
std::vector<saga::url> entries = dir.list();

for ( std::size_t i = 0; i < entries.size (); ++i )
{
    std::cout << " " << entries[i].get_string ();

    if ( dir.is_dir (entries[i]) )
        std::cout << "/";

    if ( dir.is_link (entries[i]) )
    {
        std::cout << " -> ";
        std::cout << dir.read_link (entries[i]).get_string ();
    }

    std::cout << std::endl;
}
```

api example: saga-file cat

[tools/clutils/file/cat_impl.cpp]

```
saga::filesystem::file f (saga::url(fileURL), saga::filesystem::Read);

while ( true )
{
    saga::size_t const n = 1024*64;
    saga::uint8_t data[n+1];

    for ( unsigned int i = 0; i <= n; ++i ) { data[i] = '\\0'; }

    // read a chunk into the buffer
    if ( f.read (saga::buffer (data, n), n) )
        std::cout << data;
    else
    {
        break;
    }
}
```

api example: saga-file remove

[tools/clutils/file/cat_impl.cpp]

```
saga::filesystem::file file (saga::url(fileURL), saga::filesystem::ReadWrite);  
file.remove();
```

api example: saga-advert

[tools/clutils/advert/*.cpp]

```
saga::url ad(advertURL);  
advert::entry e(ad, saga::advert::Create);
```

```
saga::url ad(advertURL);  
advert::entry e(ad);  
  
std::vector<std::string> keys(e.list_attributes());  
if (!keys.empty()) {  
    std::for_each(keys.begin(), keys.end(), print_attributes(e));  
}
```

```
saga::url ad(advertURL);  
advert::entry e (ad, saga::advert::ReadWrite);  
  
e.set_attribute(key, val);
```

api example: saga-replica

[tools/clutils/replica/*.cpp]

```
saga::url lfn(lfnURL);  
replica::logical_file logfile (lfn, saga::replica::Write);  
  
logfile.add_location(pfnURL);
```

```
saga::url lfn(lfnURL);  
replica::logical_file logfile (lfn);  
  
std::vector<saga::url> locations(logfile.list_locations());  
std::vector<saga::url>::const_iterator it;  
if (!locations.empty()) {  
    for(it = locations.begin(); it != locations.end(); ++it)  
    {  
        std::cout << " " << (*it) << std::endl;  
    }  
}
```

api example: saga-job

[tools/clutils/job/job_submit_impl.cpp]

```
saga::url js_url(service_url);

saga::job::description jd;
jd.set_attribute (saga::job::attributes::description_executable, exe);
jd.set_attribute (saga::job::attributes::description_interactive,
                  saga::attributes::common_true);
jd.set_vector_attribute (saga::job::attributes::description_arguments,
                          argvec);

saga::job::service js (js_url);
saga::job::job job = js.create_job (jd);

// create io streams for job io
saga::job::ostream in  = job.get_stdin ();
saga::job::istream out = job.get_stdout ();
saga::job::istream err = job.get_stderr ();

job.run ();

...
```

api example: saga-job (I/O)

[tools/clutils/job/job_submit_impl.cpp]

```
...  
  
while ( true )  
{  
    char buffer[1024*64];  
  
    // get stdout  
    out.read (buffer, sizeof (buffer));  
    if ( out.gcount () > 0 )  
        std::cout << std::string (buffer, out.gcount ()) << std::flush;  
  
    // get stderr  
    err.read (buffer, sizeof (buffer));  
    if ( err.gcount () > 0 )  
        std::cerr << std::string (buffer, err.gcount ()) << std::flush;  
  
    if ( out.fail () || err.fail () )  
        break;  
}
```