Proposal Code Prep Suggestions

Java Package Names

It is a best practice to have your java package names be org.opendaylight.<anticipated project repo name>. So for a project Foo, you would use packages under org.opendaylight.foo.

Copyright and license headers in source code files

Please make sure all source code files contain a correct header of the form:

```
/*
 * Copyright (c) <year> <copyright holder> and others.  All rights reserved.
 *
 * This program and the accompanying materials are made available under the
 * terms of the Eclipse Public License v1.0 which accompanies this distribution,
 * and is available at http://www.eclipse.org/legal/epl-v10.html
 */
```

Where you substitute the year and copyright holder. You can see an example in the code at the top of the file here| here. (Note: Please make sure to use the correct copyright holder rather than copying verbatim the example ;)).

Please note: when subsequent contributors make substantive changes to a file they may also optionally add a copyright header for themselves, but should preserve existing copyright statements and the license.

Example:

```
/*
 * Copyright (c) 1970 Yoyodyne and others.  All rights reserved.
 * Copyright (c) 2013 John Smith
 *
 * This program and the accompanying materials are made available under the
 * terms of the Eclipse Public License v1.0 which accompanies this distribution,
 * and is available at http://www.eclipse.org/legal/epl-v10.html
 */
```

See Developer_Best_Practices for more details as well.

Debranding code

Try to root out references to particular companies, trademarks, or brands in the code. Note, this does not apply to normal copyright headers referencing a company as the copyright holder.

OSGI and Maven Repo

Java components intended to run in process with the controller should be OSGI bundles so they can 'plug in' reasonably. You need to be able to push your binary artifacts to the Nexus repo for consumption... this is most easily done using maven as your build system.

Maven dependencies

OpenDaylight maintains a number of caching proxies for maven repos at nexus.opendaylight.org. Ideally, we would like to see all artifacts for the build come from there, so if you need artifacts that are *not* available there, please ask to have a caching proxy for the repo you do need to be added to nexus.opendaylight.org by going to https://support.linuxfoundation.org. Ideally, all repository entries in OpenDaylight POM files should refer to a nexus.opendaylight.org repo.

Try to use common services and follow common conventions

Commons services

OpenDaylight already provides common services for things like clustering storage (see for example | clustering services). Please try to use existing services whenever possible, and to complain about unmet needs rather than simply reproducing services needlessly.

Modularity conventions

It is a common modularity convention to have a package org.opendaylight.foo containing interfaces and then an org.opendaylight.foo.implementation containing the particular implementation, this pattern allows for clean separation of interface from implementation at the bundle level, and allows people to bring in other implementations simply later.

It is a common modularity convention to have a package org.opendaylight.foo.northbound for the additional details needed to expose an interface northbound via REST or other IPC.