

Extension-Based Build Configuration

The OpenCable Application Platform provides a base specification along with several optional extensions. The OpenCable Reference Implementation provides mechanisms for creating various build configurations depending on which extensions are targeted for a particular deployment. The build system will only include the binary and resource files that are required for a particular configuration. This allows deployment teams to minimize the binary footprint of the RI on resource-constrained devices.

Supported Extensions and Configurations

The RI currently implements the following OCAP extensions:

- DVR
- Home Networking
- Front Panel
- Device Settings

The following extension configurations are tested and supported by the RI development team.

Extension	Configuration 1 (None)	Configuration 2 (DVR-Only)	Configuration 3 (Non-DVR)	Configuration 4 (HN-Only)	Configuration 5 (DVR+HN)	Configuration 6 (Full)
DVR	No	Yes	No	No	Yes	Yes
Home Networking	No	No	Yes	Yes	Yes	Yes
Front Panel	No	No	Yes	No	No	Yes
Device Settings	No	No	Yes	No	No	Yes

Enabling / Disabling Extensions

You can enable or disable the inclusion of various extensions by modifying the following section in **`$OCAPROOT/target/$OCAPTC/buildrules.properties`**:

```
#
# Enable/Disable build of OCAP Extensions
#

# DVR Extension
build.ocap.DVR_ENABLED=false

# Home Networking Extension
build.ocap.HN_ENABLED=false

# Front Panel Extension
build.ocap.FP_ENABLED=true

# Device Settings Extension
build.ocap.DS_ENABLED=true
```

Any time you change the extension configuration, ensure that you perform a "clean purge build" of the system to ensure that only the desired extension binaries are included.

In addition to the build system configuration, the RI contains a new "properties management" implementation that allows developers to pick and choose various pieces of information from Java properties files based on enabled extensions and extension precedence. Details about the property management system can be found [here](#).