

```
/*
** RELEASE STATEMENT(s):
**
**             UNLIMITED RIGHTS
** The Government has the right to use, modify, reproduce, release, perform,
** display, or disclose this application programmable interface in whole or in
** part, in any manner and for any purpose whatsoever, and to have or
** authorize others to do so.
**
** Distribution Statement A - Approved for public release; distribution is
** unlimited (27 August 2015).
*/

/*
** JTNC Standard:
** Software Communications Architecture
** Appendix C: Core Framework Interface Description Language (IDL)
** Version: 4.1, 20 August 2015
*/

//Source file: CFLoadableInterface.idl

#ifndef __CFLOADABLEINTERFACE_DEFINED
#define __CFLOADABLEINTERFACE_DEFINED

#include "CFFileSystem.idl"
#include "CFPlatformTypes.idl"

module CF {

    /* This interface provides a device with software
       loading and unloading behavior. */
    interface LoadableInterface {

        /* This enumeration defines the type of load to be performed.
           The load types are in accordance with the code element
           within the softpkg element's implementation element. */
        enum LoadType {
            KERNEL_MODULE,
            DRIVER,
            SHARED_LIBRARY,
            EXECUTABLE
        };

        /* This exception indicates that the device
           is unable to load the type of file designated by the
           loadKind parameter. */
        exception InvalidLoadKind {
        };

        /* This exception indicates that an error occurred during
           an attempt to load the device. The message provides additional
           information describing the reason for the error. */
        exception LoadFail {
            CF::ErrorNumberType errorNumber;
            string msg;
        };

        /* This operation provides the mechanism for loading software
           on a specific device. The loaded software may be subsequently
           executed on the device, if the device is an executable device. */
        void load (
            in CF::FileSystem fs,
            in string fileName,
            in CF::LoadableInterface::LoadType loadKind
        )
        raises (CF::InvalidState,
```

```
        CF::LoadableInterface::InvalidLoadKind,
        CF::InvalidFileName,
        CF::LoadableInterface::LoadFail);

    /* This operation provides the mechanism to unload software
       that is currently loaded. */
void unload (
    in string fileName
)
    raises (CF::InvalidState,
           CF::InvalidFileName);
};
#endif
```