Changes to watch in the NDFS-II API

This document is intended to users who already use the data flow system. If you never use it, you can discard this document.

The top-level API changed to increase consistency and now all methods follow consistent guidelines. Before some methods returned **0/-1**, others returned **true/false** with some of them following the Java guidelines for naming convention with others the C/C++ convention.

In the new API all methods returning their status of successful execution return **true/false**. The Java and C naming convention are both followed.

The existing clients using the NDFS-II library should all be rebuilt against the new library.

This document provides a list of methods having a prototype change that wouldn't trigger any compilation error.
The user should track these methods in his code to ensure it will keep behave in the proper way.

Here is an example of a method that changed.

```
if ( sf->connect_to_application_server() != 0 )
```

becomes

```
if ( sf->connect_to_application_server() != true )
```

Following is a list of methods that changed. All these methods used to return an **integer** for status. They now return a **boolean**.

Class Smartflow

```
int init(int &argc, char** &argv, bool keepSFOptions = false,
         const std::string application=DEFAULT_APPLICATION_NAME);

int destroyFlow( Flow * flow )

int connect_to_application_server(const string clientname=DEFAULT_EMPTY,
                                  const string clientgroup=DEFAULT_CLIENT_GROUP)

int disconnect_from_application_server(void)

int startClient(const std::string clientpath, const std::string server,
                const std::string arguments=DEFAULT_EMPTY,
```

```
                const std::string parameters=DEFAULT_EMPTY)

int stopClient(const std::string clientname,const std::string clientgroup)

int getApplicationDescription(std::string &flowlist)

int Smartflow::getServerList(std::string &serverlist)

int Smartflow::getLastModifications(std::string &lastmodifications)

int pauseFlow(const std::string &clientname,const std::string &clientgroup,
              const std::string &flowtype,const std::string &flowgroup)

int restartFlow(const std::string &clientname, const std::string &clientgroup,
                const std::string &flowtype, const std::string &flowgroup)

int set_user_exit_function(void (*user_exit_function_) (), bool controlC = true)

int set_user_exit_function(void (*user_exit_function_) (void*userarg),
                           void *arg, bool controlC = true)
```

Class Buffer

```
int init(MemoryManager *manager,Flow * flow)

int init_buf(void)

int clear(void)

int set_size(buffer_size_t new_size)
```

Class ConnectionManager

```
Int init(Smartflow *sf)

int connect_to_application_server(const std::string &name,
                                  const std::string &group)

int disconnect_from_application_server(void)

int get_flow(const std::string &type, const std::string &name,
             const std::string &group, const std::string &arg,
             flow_direction_t direction, int blockcount, int blocksize,
             int check, unsigned int numberOfConsumersExpected)

int removeFlow(const std::string &type, const std::string &name,
               const std::string &group)
```

```
int checkFlow(const std::string &type,const std::string &name,
              const std::string &group)

int startClient(const std::string &clientpath, const std::string &server,
                const std::string &arguments, const std::string &parameters)

int stopClient(const std::string &clientname,const std::string &clientgroup)

int getApplicationDescription(std::string &flowlist)

int getServerList(std::string &serverlist)

int getLastModifications(std::string &modifications)

int pauseFlow(const std::string &clientname,const std::string &clientgroup,
              const std::string &flowtype,const std::string &flowgroup)

int restartFlow(const std::string &clientname,const std::string &clientgroup,
                const std::string &flowtype,const std::string &flowgroup)

int sendLog(const std::string &logMessage)
```

Class MemoryManager

```
int init(Smartflow *sf)
```

Class ErrorManager

```
int init(Smartflow *sf)
```

Class FlowManager

```
int init(Smartflow *sf)

int destroyFlow(Flow *f)

int stopAllFlows()
```

Class Preference

```
int init(PreferenceManager *manager)

int initPreferences(int &argc, char** &argv, bool keepSFOptions=true)
```

Class PreferenceManager

```
int init(Smartflow *sf)

int init_args(int&, char**&, bool keepSFOptions)
```

Class Flow

```
int init(FlowManager *manager,const std::string &type, const std::string &name,
         const std::string &rgroup, const std::string &args,
         flow_direction_t mode, flow_policy_t policy, int isSynchronized)

int start(void)

int stop(bool waitForEmptyHistory = false)

int releaseBuffer(Buffer * buf)

int cancelBuffer(Buffer * buf)

int pause()

int restart()
```