Okay Bay currents app code was apparently written in Objective C (hence the .m and .h files).

This is for Mac Os X machines. We may be able to translate to python? Or some other code.

Code was written by Guilherme Carvalho

There appear to be separate builds for the ipad and the iphone/OsX.

Classes directory has most of the code.

Resources has what appear to be nothing off interest

Sfsu.xcodeproj Xcode project files and possibly compile files?

Main.m uses UIKit apple developer interface kit for apps. Note there is something else called UIKit on github which appears to be something else though possibly related.

Entitlements.plist is an xml document referencing an apple dtd.

This defines primitive types “String”, “data”, “date”, “true”, “false”, “real” and “integer”.

MainWindow.xib this is supposed to be associated with and UIViewController. The file is an xml document. The document contains the layout of the interface. May need some apple code to actually see what this looks like. Frame size is [320, 480]

Classes:

AboutView: Creates the about page on the app. Has a link to <http://www.norcalcurrents.org/COCMP/Real-Time> Currents.html

AQXMLParser: XML parser

AQXMLParserDelegate: More of the XML parser package

ArrowAnnotation: Uses red and green to annotate an arrow (not ADA compliant for colorblind)

Buoy: Draws an scaled arrow based upon the u,v data

ByteUtils: Read data from binary array, various utilities to do this.

CacheManager: Manage cache.

CDataScanner: various data conversion loader, appears to work with pointer location data.

CDataScanner\_Extensions: Access HTML CDATA objects and parse out comments

CJSONDataSerializer: Serializes data and escapes some characters for ingest.

CJSONDeserializer: Unpack a serialized data array.

CJSONScanner: Scan JSON file and check for errors.

CJSONSerializer: Serialize CJSON for data transfer?

ColorUtils: Work with colormap Extract RGBAlpha where Alpha is set to FF (opaque).

Constants: Various Constants related to code not to data placed here.

CSerializedJSONData: Code to serialize JSON data.

Datasource: General functions to deal with functions and creating strings to use?

DatasourceManager: More data names and get .k files

DDAnnotation: Working with initializing Data Coordinates.

DDAnnotationView: Pins and other animations and shadows.

iPhoneNonatomic: define iphone or not.

MapView: Lat, Lon 37.87, -122.417 center. Span lat=0.09 deg, Span long=0.189. Add lots of map functions and buttons.

NSCharacterSet\_Extensions: Define some character codes.

NSDataBase64: Define more characters.

NSDictionary\_JSONExtensions: Extend JSON with error data added.

NSScanner\_Extensions: More scanner to check for errors.

ResourceManager: Mange resources.

RMSTracker: Load and unload data files and locations (no actual path or locations here).

SeaFile: allocate and initialize data array (called buoys).

SeaSpeed: allocate and initialize files array?

SfsuAppDelegate: After finish launching stuff and deallocation stuff.

SfsuViewController: Looks like this might actually be the data load? [http://www.jumpfox.com:8080/sfsu/gl and /data\_sea.zip](http://www.jumpfox.com:8080/sfsu/gl%20and%20/data_sea.zip) and data\_sea.xml jumpfox.com doesn’t exist anymore!!

SfsuXMLParser: Returns seaSpeed array of data.

SizePositionConstants: Sets up location of Frames in app.

StreamUtils: Determine data stream status

TextUtils: Encode, decode, and other string funtions.

TouchXML: load various headers.

TreeNode: Create tree node.

UITabBarController: Create Tab Bar Controller

XMLParser: Parse XML string