

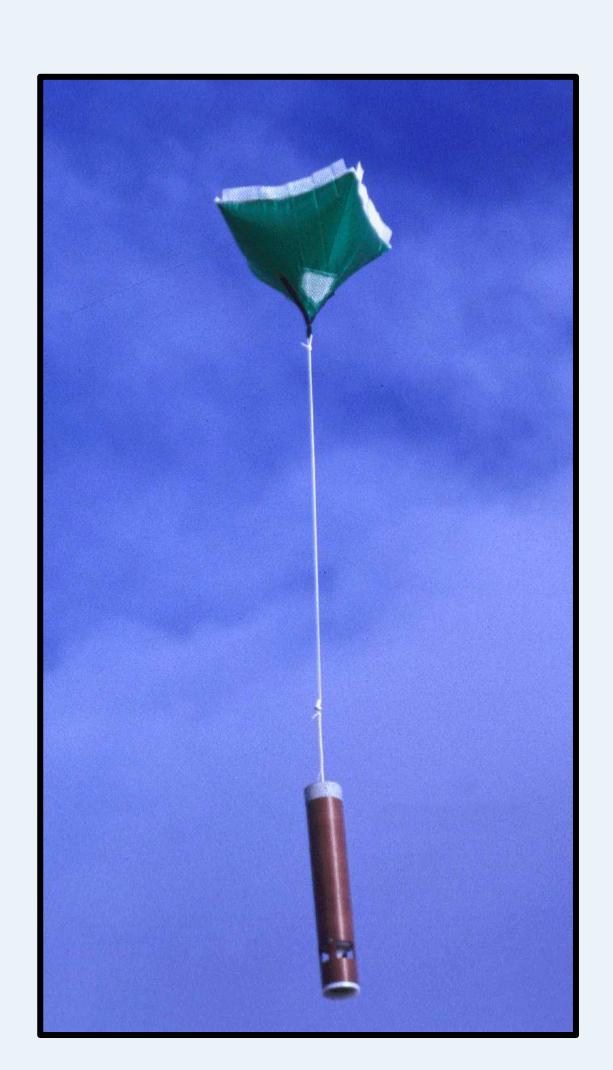


Author: Floriana Ciaglia

## INTRODUCTION

ASPEN (Atmospheric Sounding Processing Environment) is an application used for analysis and quality control (QC) of sounding data. Sounding data is a profile of the atmosphere that is collected by an instrument that rises, a upsonde, or falls, a dropsonde, through the atmosphere. Aspen is commonly used for collecting data to provide hurricane forecasts.

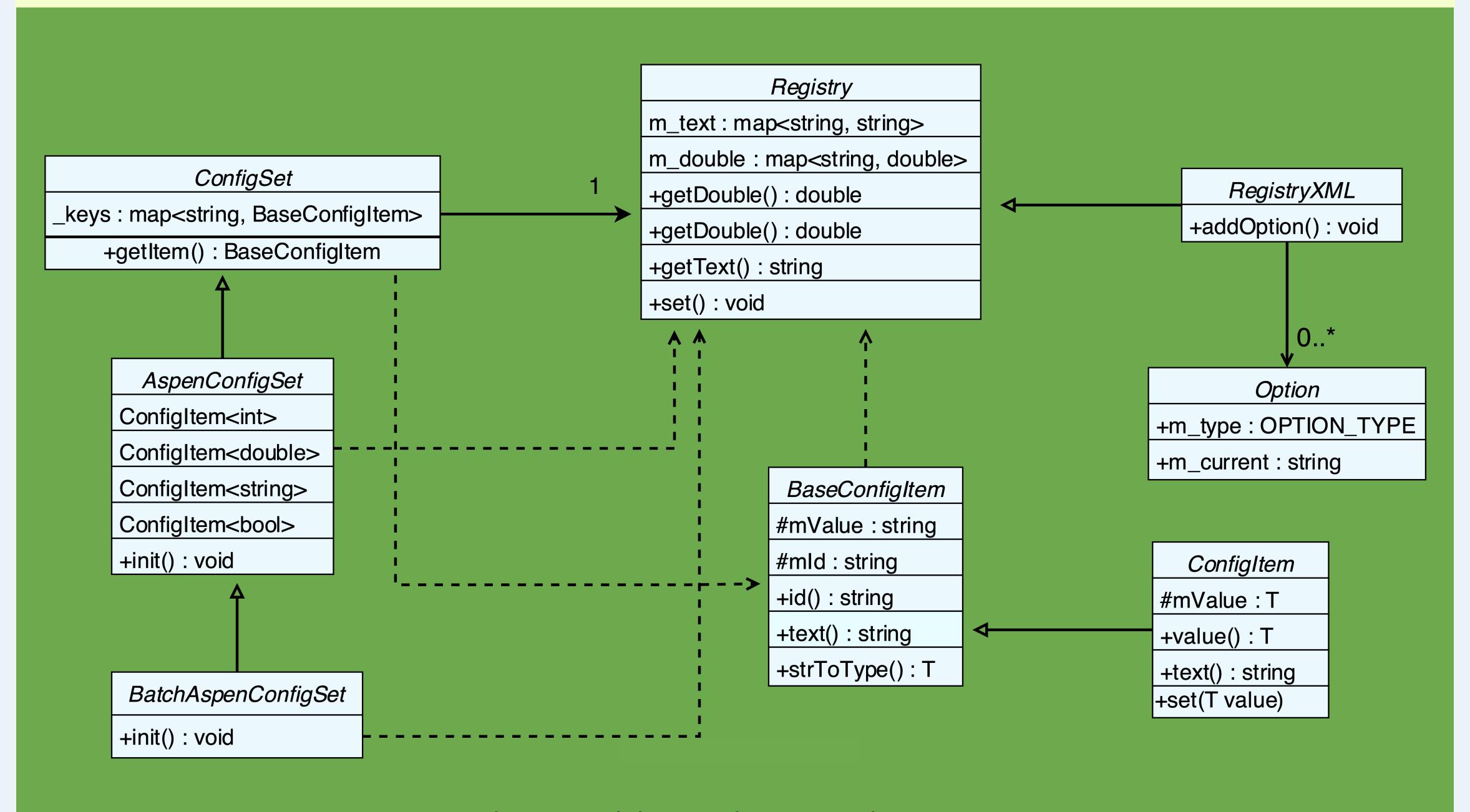
ASPEN has been in development since the early nineties. Thus, many of the coding conventions used to write it are outdated. This project focused on refactoring the architecture of the configuration objects, which involved modifying pieces of code to work better with more recent coding practices. By updating the architecture of these objects, our goal was to simplify future development of the application and improve software longevity.



GPS dropsonde collecting sounding data.

# Aspen Config-Set Refactor

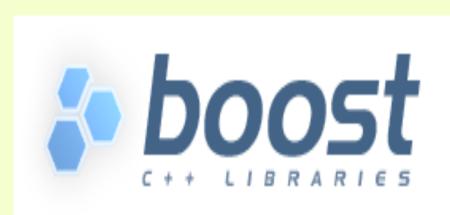
Apply improvements to older coding practices with the goal of more sustainable software



UML diagram of the Configuration directory in Aspen.

## **TOOLS**







## **ACKNOWLEDGEMENTS**

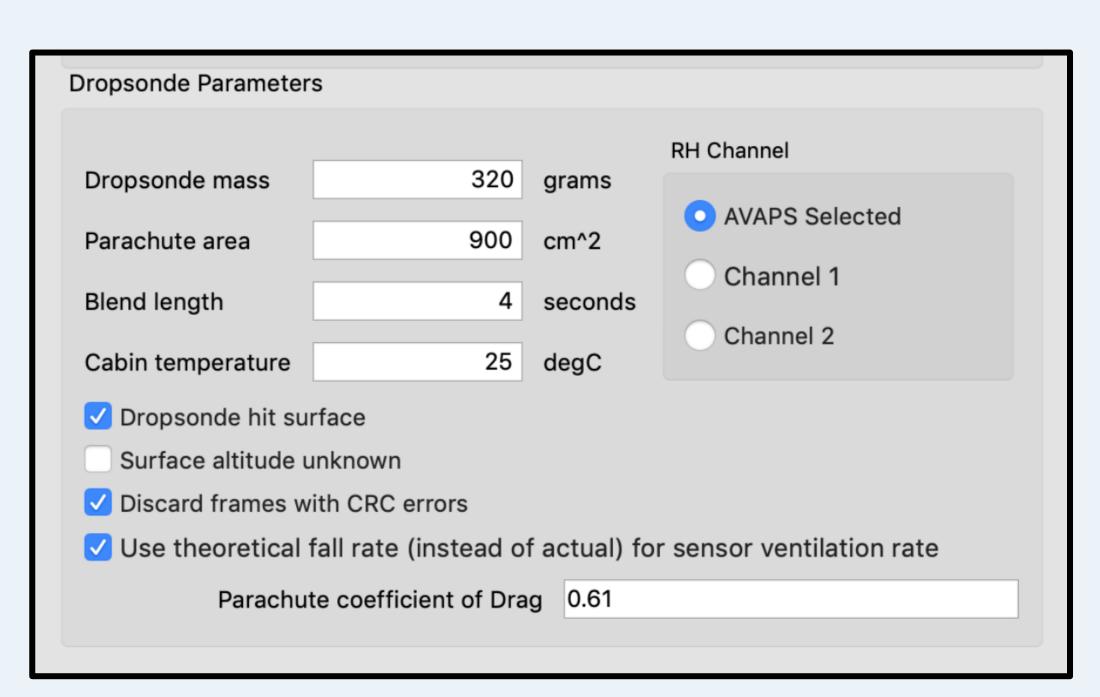
I would like to thank my mentor, Isabel Suhr, the Earth Observatory Laboratory (EOL) staff and the SUPER program coordinators for their help and support throughout this internship experience.

## **METHODS**

ASPEN uses a set of configuration parameters, called Configltems, that can be modified and saved by users.

Main changes that the refactoring involved:

- the templatization of the Configltem object so that it can be used to store values of different types. This allowed us to decrease the verbosity of the code and to reach a higher level of versatility by having the same class function with all primitive data types.
- standardizing the way ConfigItem metadata was being modified. We removed a redundant way to change saved information, which prevents inconsistencies between the configuration information in memory and the one saved on disk.



Example of Configuration items in Aspen UI.

## **RESULTS**

This project has started the refactoring process of the ASPEN configuration objects. Throughout the SUPER program we were able to achieve improvements that relate to the templatization of the Configltem object, the modification of Configltem metadata and other smaller changes. Some of the future goals include:

- changing the configuration language to a more human readable one.
- implementing inheritance in ConfigSet in order to remove redundancy in the configuration file.