

Settings Manager

Введение

Соглашения о терминах

Область применения

Применяется для предоставления сервиса по работе с конфигурациями в распределенной системе обработки информации и управления радиолокатором.

Общее описание

Назначение и возможности продукта

Менеджер конфигураций предоставляет сервис по загрузке\сохранению и работе с отдельными параметрами конфигурации. Конфигурации хранятся в файлах в формате JSON.

Среда функционирования продукта

Исполняется в среде операционных систем Windows и Ubuntu Linux. Модуль поставляется в качестве исходных текстов – заголовочные (.h) и компилируемые (.cpp) файлы или статической библиотеки – заголовочный (.h) файл и объектный код (.lib и .a) файл.

Допущения и зависимости

При реализации используются библиотеки:

- Boost::filesystem
- rapidjson

Описание системы

Файлы конфигурации хранятся в формате JSON (версии RFC 7159) и имеют расширение .cfg. Имена файлов без расширения соответствуют имени конфигурации, т.е, например, при имени файла «config001.cfg» – имя конфигурации «config001». При загрузке конфигурации она становится *текущей* и все манипуляции по изменению параметров производятся с текущей загруженной конфигурацией в памяти менеджера, не затрагивая эту конфигурацию на диске.

Файл конфигурации разбит на *категории настроек* – вложенные объекты, которые описывают конфигурацию отдельных модулей системы обработки информации и управления радиолокатором. Организация и разбор внутренней структуры каждой из категории настроек модулей ложится на разработчиков этих модулей.

Пример файла конфигурации:

```
{
  "category1" :
  {
    "key1" : "value1",
    "key2" : "value2",
    "key3" : "value3",
    "array1" : [ "value1", "value1", "value1" ]
  },
  "category2" :
  {
    "key1" : "value1",
    "key2" : "value2",
    "object1" :
    {
      "nested_key1" : "nected_value1",
      "nested_key2" : "nected_value2",
    }
  },
  ...
}
```

При вызове интерфейсных функций возможны следующие типы исключений:

- SystemException – ошибки, вызванные ошибками при работе с функциями операционной системы или сторонних библиотек.
- ConfigException – ошибка при чтении или разборе описания конфигурации.

В случае вызовов функций чтения или записи отдельных параметров до загрузки какой-либо конфигурации генерируется исключение ConfigException.

Более детальное сообщение об ошибке выводится в файл <Дата и время запуска>.log в каталоге <Исполняемый файл>/logs/.

Функциональные требования

- set_path(dir_path).
Указывает папку с файлами конфигурации. В случае ошибки генерирует исключение SystemException.
- load_config(config_name).

Загружает конфигурацию config_name (см. раздел Описание системы) в качестве текущей. Если предварительно ранее была загружена другая конфигурация, то при удачной загрузке произойдет замещение конфигурации, при неудачной старая конфигурация сохранится. В случае ошибки генерирует исключение SystemExeption (если конфигурация отсутствует) или ConfigExeption.

- save_config(config_name).
Сохраняет текущую конфигурацию под именем config_name. В случае наличия конфигурации с таким именем происходит **перезапись** конфигурации. В случае ошибки генерирует исключение SystemExeption или ConfigExeption.
- update_config(config_description).
Полностью обновляет (**перезаписывает**) текущую конфигурацию согласно config_description (JSON). В случае ошибки генерирует исключение ConfigExeption и сохраняет предыдущую конфигурацию.
- String config_name().
Возвращает имя текущей конфигурации. В случае вызова функции до загрузки какой-либо конфигурации возвращает пустую строку.
- String config_path().
Возвращает строку с расположением папки файлов конфигурации. В случае вызова функции до указания папки с конфигурациями возвращает пустую строку.
- String get_config_str().
Возвращает строку с полным описанием текущей конфигурации в формате JSON. В случае вызова функции до загрузки какой-либо конфигурации возвращает пустую строку.
- Settings get_section(section_name).
Возвращает дерево настроек settings для указанной категории настроек (см. раздел Описание системы). В случае ошибки (неизвестная категория настроек) генерирует исключение ConfigExeption.
- Object get_object(settings, key).
Возвращает вложенный объект из дерева настроек (или объекта уровнем выше) settings по ключу key. Объекты могут представлять собой типы JSON: Object и Array. В случае ошибки (неверный ключ или объект по ключу не является Object или Array) генерирует исключение ConfigExeption.
- Value get_value(settings, key).
Возвращает параметр типа Value из дерева настроек или объекта settings по ключу key. Функция должна обеспечивать работу с типами : int32, uint32, int64, uint64, double, string. В случае ошибки генерирует исключение ConfigExeption.
- Array get_array(object).
Возвращает массив из объекта object. Функция должна обеспечивать работу с массивами типа : int32, uint32, int64, uint64, double, string. В случае ошибки генерирует исключение ConfigExeption.
- set_value(settings, key, new_value).
Перезаписывает параметр из дерева настроек или объекта settings по ключу key значением new_value. Функция должна обеспечивать работу с типами : int32, uint32, int64, uint64, double, string. В случае ошибки генерирует исключение ConfigExeption.
- set_array(object, new_value).
Перезаписывает массив из объекта object значением new_value. Функция должна обеспечивать работу с массивами типами : int32, uint32, int64, uint64, double, string. В случае ошибки генерирует исключение ConfigExeption.