aCIBRunShellRunner

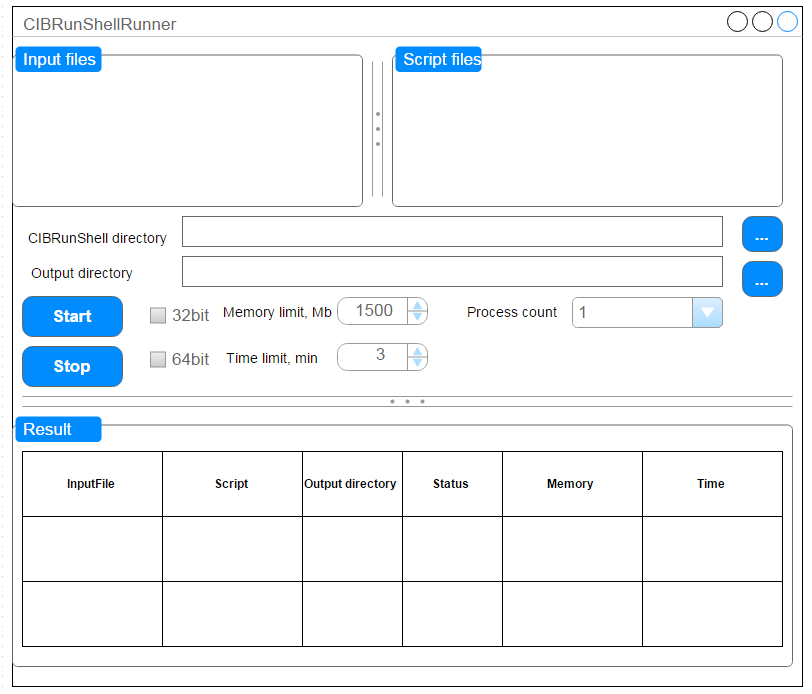
## Постановка задачи

Реализовать приложение для автоматизированной обработки файлов программой CIBRunShell. CIBRunShell – консольное приложение для обработки и преобразования различных файлов. Для управления CIBRunShell-ом могут использоваться аргуметы командной строки, ini и xml файлы. Цель нового приложения - упростить и ускорить процесс обработки файлов с использованием заранее созданных управляющих файлов(сценариев).

## Основные требования к реализации

* .net Framework 4.5
* WPF c использованием ICommand и Binding
* Асинхронная обработка входных файлов
* Возможность многопоточной обработки входных файлов

## Интерефейс пользователя



**Input files** – список входных файлов.

Возможные варианты добавления файлов:

* Перетаскиванием из проводника
* Через контекстное меню – Add files

Возможные варианты удаления файлов:

* Через контекстное меню Remove files
* Клавишей Delete

Возможно добавление файлов с любым расширением или по предустановленным фильтрам:

Documents(pdf, rtf)

Images(png, tiff)

**Script files** – список сценариев обработки. Управление списком аналогично управлению списком файлов.

Допускается добавление только следующих файлов: ini, xml, arg, где arg это текстовый файл содержащий параметры командной строки

**CIBRunShell directory** – папка где находятся исполняемые файлы CIBRunShell(. Может быть выбрана нажатием на кнопку “…” или путь вставляется непосредственно в текстбокс.

**Output directory** – папка куда будут записаны результаты обработки. Может быть выбрана нажатием на кнопку “…” или путь вставляется непосредственно в текстбокс.

**32bit/64bit**  разрядности CIBRunShell которые должны быть использованы

**Memory limit** – ограничение на использование памяти. Если процесс CIBRunShell-a использует более указанного значения - он должен быть остановлен

**Time limit** – ограничение на время работы. Если процесс CIBRunShell работает больше указанного времени он должен быть остановлен.

**Process Count** – кол-во одновременно работающих процессов CIBRunShell

**Start** – кнопка запуска процесса обработки

**Stop** – кнопка прекращения процесса обработки

**Result** – журнал обработки. По даблклику на строке в эксплорере должна открыться папка “Output directory”

Status – текущий статус обработки указанного файла указанным сценарием. Возможные значения: Waiting, Running, Completed, Failed, KilledByMemory, KilledByTime. Разные статусы должны иметь разный цвет текста в ячейке.

Memory – максимальное кол-во памяти использованной процессом CIBRunShell-a

Time – время работы CIBRunShell-a.

Во время обработки файлов все управляющие элементы кроме кнопки Stop должны быть заблокированы.

Все настройки кроме списков входных файлов и сценариев должны сохраняться между запусками.

В строке статуса во время выполнения должно выводиться общее время обработки, общее кол-во элементов для обработки, кол-во обработанных элементов.

## Общий принцип обработки файлов

После нажатия на кнопку старт необходимо

1. Для каждого набора входной файл/выходной файл/разрядности сформировать запись в журнале обработки
2. Для каждой записи выполнить запуск CIBRunShell соответствующей разрядности с передачей параметров указанных в сценарии. При этом для каждой записи должна быть создана директория с уникальным именем в директории Output dir. В данной директории должны быть созданы директории Input и Output. В директорию Input необходимо скопировать файл сценария с подставленным значениями подстановочных символов, а так же создать тектовый файл с указанием имени входного файла и разрядности.
3. Результаты обработки отобразить в таблице

## Подстановочные символы

Т.к. сценарии содержат пути к входным/выходным файлам и файлам настройки то необходимо предусмотреть замену подстановочных символов на реальные значения перед запуском сценария. Т.е. $InputFile$ должно быть заменено на полный путь ко входному файлу, $OutputDir$ должно быть заменено на путь к подпапке Output.

Другие подстановочные символы могут быть добавлены в файл replace.cfg в виде: $key$=value. Одна строка – одно определение. Предполагается что файл replace.cfg находится в той же директории что и запускаемый файл. Файл replace.cfg может отсутствовать и это не считается ошибкой.