# 虚拟磁盘需求分析

## 基本需求：

* 实现磁盘功能，能够管理基础文件、文件夹、符号链接文件。
* 实现类windows CMD 功能，能够对磁盘数据进行操作。
* 实现文件读写器，能够通过读写器将虚拟磁盘数据保存为一个文件，别且能够通过读取文件初始化磁盘。
* 接入白盒测试并且提供至少5个用例进行测试。
* 磁盘需要实现多盘符操作，并且提供创建新盘符的功能。并约定磁盘符号为类windows（C:、D:、E: ... ...）。
* 项目需要接入白盒测试模块，则根据白盒测试要求，需要存在VirtualDisk类以及相应接口。
* 磁盘操作命令，在下文进行详细叙述。
* 在操作磁盘时，输入路径需要支持通配符\*（0~n）以及？（1）。

## **模块设计及难点约束：**

* 整体设计如下（下图不包括白盒测试接入相关）：



* 整体操作执行流程：

用户输出信息到输入模块，输入模块预处理之后较为命令模块，命令模块根据输入内容，调用磁盘操作模块，磁盘操作模块通过组合调用数据磁盘模块底层接口完成功能要求，返回结果给命令模块，命令模块根据对应命令输出格式输出结果给用户。这里没有提供输出方式的重定向，适应性稍差。

* 磁盘系统：



磁盘系统分为两部分：数据磁盘模块以及磁盘操作模块，

* 数据磁盘模块（提供底层操作）

数据磁盘模块来维护磁盘数据以及提供底层基础接口。将数据以一颗树的形式存储。并且提供添加文件，删除文件，创建文件，销毁文件，查询文件存在的接口。

* 磁盘操作模块（提供上层操作）

磁盘操作模块用来调用数据磁盘提供的基础接口，来实现命令需要的复杂操作与命令模块进行直接交互以及调用文件读写器进行磁盘数据持久化以及导入数据功能，如移动，拷贝，删除，创建文件，验证文件是否存在，格式化磁盘，导入、导出文件。

磁盘操作模块具体承担责任，通配符路径通配，创建初始化文件、磁盘、文件夹、符号链接，以及一些验证。

* 符号链接文件

磁盘内由于需要维护文件数据，为了避免当符号链接文件所指向的文件被删除或者被移动，导致符号链接失效导致不可预知的结果，作如下设计，当所只想文件被删除时，将把所有指向该文件（或文件夹）的符号链接文件标记为无效，不做其它处理。此时在对符号链接文件进行操作，则提示错误。具体符号链接文件相关下文做重点叙述。

* 文件操作系统



文件操作系统主要分为四个部分：系统模块，输入模块，命令工厂，命令模块。

* 系统模块

模块做系统保持loop，将用户输入信息，传出输入模块进行预处理，接受处理结果将结果传入命令工厂实例命令对象，再将实际参数出入命令对象，调用执行接口，并且需要保持当前工作路径（以字符串的形式）。

* 输入模块

输入模块主要职责为将传入字符数据进行预处理，整理出命令名字以及参数列表（包括功能参数和路径信息）。处理数据后将数据包返还给操作模块。需要注意的此处分割信息以空格为分割符并向下预读一位，原因是文件的明明中间需要包含空格，则包含空格时名字以双引号包裹。所以找到空格之后向下预读一位看是否时“符号。并且将/t 字符做错误处理。即输入内容中不允许包含/t。

输入预处理：将输入字符串直接处理为一个结构体成员包含命令名，参数列表，路径列表。

处理规则：遍历路径如果遇到空格且没有被“”包含则分隔，优先填充命令名，如果命令名已经被初始化则看字符串是否以‘/’为开始且没有路径信息被初始化如果是则被识别为参数。

* 命令工厂

根据系统模块传入内容实例命令对象返回给系统模块。、

构造时将初始化所有命令对象，当调用获取命令对象时对命令对象进行reset，然后返回。（避开之前设计重复申请释放 命令不能够被重用的问题）。

* 命令模块

具体支持命令如下：

据系统模块传入参数，执行具体功能

所有命令统一路径约束：

命名约束，无论是文件或文件夹不约定长度，但是限制内容，命名允许包含 大小些字母 下划线，数字以及“.”，但不允许只包含“.”，允许出现空格但开头结尾不允许是空格，并且需要用英文双引号包含，如过是实际磁盘需要以@开头，其他字符都不允许使用。盘符只能以“[a-z]：”的形式命名

路径匹配设计：

由于路径需要匹配通配符而且又要能够处理正常路径，现做一下设计：

1. 路径需要进行初步处理，将路径中的. .. 已经其他相对路径相关，转换称绝对路径，此时路径中更有可能存在通配符。
2. 将路径按‘/’分隔为多级，即为将字符串分隔为字符串数组
3. 然后逐级查询，将每一级转化为正则表达式，用父级对象获得所有孩子对象进行匹配，这样能够把通配符一并完成通配能够统一流程。

通配符处理流程：

1. 将路径信息无论是否含有通配符进行分割每一级路径是数组中的一个元素，
2. 进行单级匹配如果当前能够匹配就继续递归，如果不能就跳出。
3. 匹配采用将传入路径的当前级生成为正则表达式。然后对当前路径进行匹配。如果能够陪到的最后一级的就是满足条件的路径。