

Home file organizer

Zápočtová práce

Filip Havel

Úvod

Tento program slouží, jako manager dat v lokální síti, poskytující přehled o jednotlivých souborech, které byly do programu načteny. Aplikace je umožňuje, především správu filmových a seriálových souborů. Hlavním úkolem aplikace je zajistit, že uživatel má přehled o tom, která data jsou kde uložena a může si dělat k souborům poznámky, které se mu přetahují do všech připojených zařízení.

Programátorská část

Program je napsán jazyce C# pro platformu .NET, uživatelské rozhraní je tvořeno pomocí knihovny WinForms. Aplikace je myšlenkově rozdělena do 4 závislých celků podle typu operace, kterou daná část vykonává. Jedná se o práci se soubory (třída FileManager), synchronizace (třída Synchronization), síťová komunikace (třída Connections) a uživatelské rozhraní (třída Form1 a další soubory ve složce GUI).

Data, jež aplikace využívá, se načítají do struktury (viz níže) a ukládají se jako xml soubory do složky, specifikované v souboru HomeFileOrganizer.exe.config.

Pokud se v dalším textu programátorské části je popis zpráv nebo textu využívaného programem, mohou v nich být části uvozené do složených závorek, tyto části včetně závorek reprezentují proměnnou informaci, která je tímto popsána.

FileManger

Tato část řeší všechno, co nějakým způsobem pracuje se souborovým systémem počítače. Zde lze tedy najít metody pro přidání souboru z lokálního disku do aplikace, přesun souboru mezi počítači (úzce spjata se síťovou částí, která se postará o to, aby byl soubor stažen z připojeného zařízení).

Dále je popis jednotlivých operací.

Přidání souboru

Přidání souboru může proběhnout ze dvou důvodů, buď na tomto počítači přidal uživatel soubor, nebo ho přidal na jiném zařízení a přidání je vyvoláno synchronizační událostí.

Pro přidání souboru, vyvolané synchronizační událostí, neprobíhá přímo skze třídu FileManger, přidá se soubor do datové třídy (HomeData), která je public položkou třídy FileManger.kde se vybere instance reprezentující disk(MyDisk) na němž je soubor uložen a zavolá se metoda MyDisk.addFile(string,Category,string). Tato metoda vytvoří dle parametru Category nový InfoFileSoubor s počátečnímy daty.

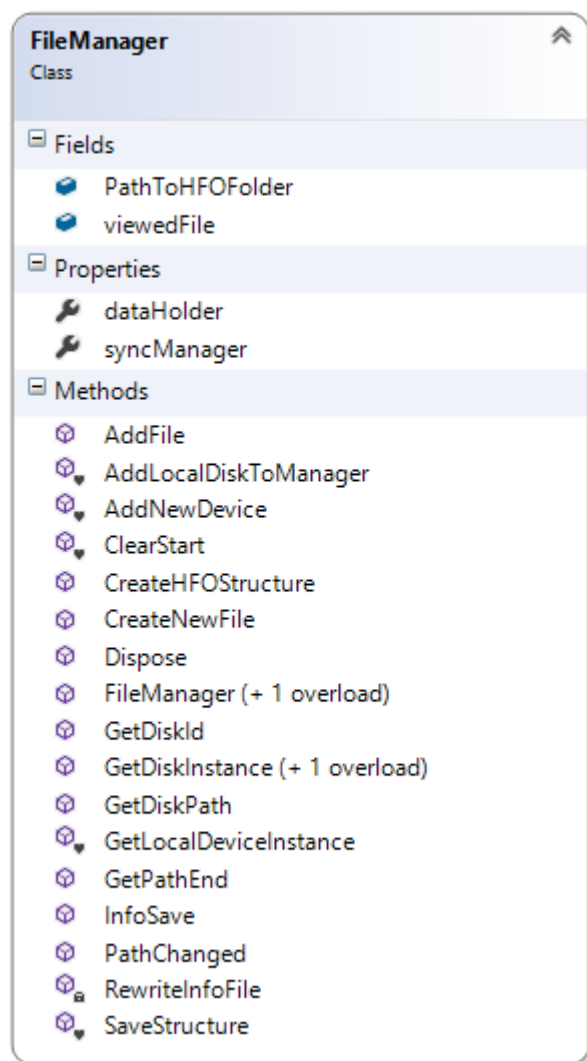
Pro přidání souboru, na tomto zařízení uživatelem, se spustí úloha pomocí metody FileManger.AddFile(string,Category). Tato metoda vytvoří Task, který vybere dle parametru cesty k souboru, správnou instanci MyDevice následně MyDisk. Na instanci MyDisk se pak zavolá metoda MyDisk.addFile(string,Category,Synchronizer), tato metoda načte skrz parametry Category počáteční stav souboru InfoFile a tyto data uloží do správné složky a nově vytvořeného souboru. Tento soubor má vygenerované jméno, tak že nepřepíše již existující soubor(metoda FileManger.CreateNewFile). K tomu také načte do InfoFile instance položky system\name (název souboru), system\size (velikost souboru v bytech), system\last chage(čas posledního zápisu do souboru) a system\creation(čas vytvoření souboru). K tomu jsou pro přidání položek vygenerovány synchronizační události FileAddEv, pro vytvoření soboru, a InfoChangeValueEv, pro zápis do výše uvedených položek.

Přidání zařízení

Přidání zařízení může proběhnout třemi způsoby.

Zprvce může byt přidání vyvoláno synchronizační událostí. V tomto případě se pouze zavolá metoda HomeData.AddDevice(ulong,string), která vytvoří novou instanci s učenými parametry a přidá ji do datové struktury, následně se přepíše soubor OverView.xml a vytvoří soubor deviceInfo(viz níže).

Zadruhé může být zařízení přidáno lokálně (v případě že vytváří síť), v tomto případě se volá metoda FileManger.AddNewDevice(), která zavolá na datové struktuře metodu



HomeData.AddDevice(string), která vytvoří instanci MyDevice s jedinečným Id a nastaví Setting.Default.localId na toto id. Následně se přidají lokální disky viz níže.

Za třetí, v případě že se zařízení připojuje do sítě, tak mu je posláno přidělené id od zařízení, které již v síti je, pak je to stejné jako první možnost. A následně se spustí přidání lokálních disků.

Přidání lokálních disků

Přidání disků může probýhat dvěma způsoby.

Za prvé jej zajišťuje synchronizační událost, v tomto případě se jen přidají do datové struktury instance s určitými parametry a k tomu se dotvoří složky v souborovém systému.

V druhém případě (přidávají se nový lokální disky) projde aplikace všechny disky dostupné na tomto zařízení, vygeneruje pro ně identifikátor, přidá je do datové struktury a vytvoří v kořeni disku soubor pro identifikaci (viz data aplikace).

Metoda PathChenged

Tato metoda zajistí přesun datové položky, mezi disky. Její účel je využití v případě, kdy se přesouvá soubor mezi zařízeními a je potřeba změnit informace o souboru v datové struktuře. Dělá to že najde instanci MyFile dle původní cesty souboru a udělá její kopii do místa určeného novou cestou k souboru, zároveň vytvoří odpovídající infoFile soubor. Pokud je parametr removeOld =true, tak starou instancí a infoFile smaže.

Další funkce

Kromě výše specifikovaných jsou ve třídě FileManager metody, který ze systémové cesty zjistí identifikátor (nebo vybere instanci) disku (resp. zařízení) v aplikaci. Také je zde metoda(CreateHFOStructure), která vytvoří soubory a složky pro aplikační data (pro použití při inicializaci sítě), nebo metody pro uložení dat (InfoSave,SaveStructure)

Data aplikace

Data jsou v aplikaci, uložena ve stromové struktuře, jejíž části lze nalézt ve složce Classes, kdy kořenem struktury je třída HomeData. Strom struktury vypadá takto:

HomeData –kořen

HomeData.devices(instance MyDevice) – seznam spravovaných zařízení

MyDevice.disks(instance MyDisk) - seznam všech spravovaných disků v zařízení

MyDisk.files(instance MyFile) - seznam spravovaných souborů na daném disku

MyFile.MyInfoFile(instance MyInfoFile) – poznámky u souboru

MyDisk.folders(instance MyRootFolder) – seznam spravovaných složek, připraveno pro možnosti rozšíření na spravování celých složek souborů, není však plně použitelné v této verzi

MyInfoFile.items(instance třídy dědicí od abstraktní třídy Item) - položky poznámek, obsahující uživatelem zapsaná data a jejich název, mohou být umístěny i instancí InfoGroup. Tyto položky se zobrazují v uživatelském rozhraní, do tabulky.

MyInfoFile.groups(instance InfoGroup, která dědí strukturu od MyInfoFile) – složky položek poznámek, které se pak v uživatelském rozhraní zobrazují do GroupBoxu s názvem uvedeným v InfoGoup.Name

Popis uložení dat

Data jsou uložena ve složce uvedené v souboru HomeFileOrganizer.exe.config (viz níže). Většina data, až na pár výjimek, se ukládají jako validní Xml soubory. Jedná se o soubory .sync, .tosync, .failsync, .xml. Výjimkou je soubor /SyncFiles/order.txt a všechny soubory, které jsou bitovými kopiemi souboru z jiných zařízení (přenesené soubory, kopie souborů).

Struktura této složky je následující:

/AppFiles – výchozí cíl pro přenos souboru. Má podsložky podle zavedených kategorií.

/DeviceInfos – obsahuje soubory s přehledy o zařízeních (typ deviceFile)

/FileInfo – obsahuje soubory s poznámkami ke spravovaným souborům. Pro každý spravovaný disk je vytvořena podsložka s názvem stejným jako je identifikátor disku

/FileInfo/{diskId}/File – obsahuje soubory typu fileInfo pro spravované soubory

/FileInfo/{diskId}/Folder – obsahuje soubory typu fileInfo pro spravované složky

/FileInfo/{diskId}/ItemFiles – obsahuje soubory, jako jsou obrázky atd., pro ukládání dat pro element.

/FITemplates – soubory se vzory informací, která mají být v souboru typu fileInfo

/SyncFiles – soubory s informacemi pro synchronizaci

/Categories.xml – soubor, v němž je definována pro každou kategorii, jaky vzory mají být použity

/OverviewFile.xml – soubor typu přehled, měl by být jen jeden na jedno zařízení

Krom těchto dat si aplikace v kořenovém adresáři každého disku vytvoří soubor „-{proměnná HFOFolderName ze souboru config }.info“, tento soubor obsahuje řádku „diskId={identifikátor disku}“, která slouží pro převod cest v aplikaci do systémové cesty a naopak.

Soubor OverviewFile.xml

```
<dataStorage>
  <computer deviceId="{identifikátor zařízení}" deviceFile=" " name="{název zařízení}"
    syncedTo="{sync soubor na zařízení, kde je tento soubor uložen, ke kterému je zařízení
      reprezentováno tímto elementem synchronizováno}" runOn="{sync soubor na zařízení, jež je
      reprezentováno tímto elementem, ke kterému je zařízení, na němž je tento soubor uložen,
      synchronizováno }">
    <disk deviceId="{identifikátor disku}" name="{název disku}"/>
    ... další elementy disk
  </computer>
  ...další elementy computer
</dataStorage>
```

Soubor typu deviceFile

```
<device>
  <disk id="{identifikátor disku}">
    <file path="{plná cesta ke spravovanému souboru, bez názvy disku (př.
      „\Users\Bubyx\Pictures\GCHD\2014-08-09\Scan10004.JPG)“}" infoFile="{jméno
      souboru typu infoFile pro tento soubor}" category="{kategorie souboru}"/>
    ...další elementy file
  </disk>
  ...další elementy disk
</device>
```

Soubor typu fileInfo

```
<file>
  <group name="{jméno skupiny, je zobrazeno v GUI}">
    <value name="{název položky, je zobrazeno v GUI}" type="{určuje jaka třída se má
      načíst, aktuální možnosti jsou text,date,link,rating,multiline,picture,counter,byte}">{hodnota
      položky}</value> //může mít dodatečné atributy
    Příklady:
    <value name="Name En" type="text">Lucy</value>
    <value name="Release date" type="date">06/08/2014 00:00:00</value>
    <value name="IMDb" type="link"
      text="Imdb">http://www.imdb.com/title/tt2872732/?ref_=fn_al_tt_1</value>
    <value name="Rating" type="rating">3</value>
```

```

    <value name="Story" type="multiline"><![CDATA[A woman, accidentally caught in a
    dark deal, turns the tables on her captors and transforms into a merciless warrior
    evolved beyond human logic.]]></value>
    <value name="Picture" type="picture">C:\Users\Bubyx\Pictures\Lucy.jpg</value>
    ...další elementy group nebo value
  </group>
  ...další elementy group nebo value
</file>

```

Soubor Categories.xml

```

<categories>
  <category name="{název kategorie}">
    <template temp="{vzor (soubor v FITemplate)}" folder="{true pokud má být přidán
    do fileInfo souboru pro složku}" file="{true odkud má být přidán do fileInfo souboru pro
    soubor}"/>
    ...další elementy template
  </category>
  ... další elementy category
</categories>

```

Soubor vzoru kategorie

```

<info>
  <group name="{název skupiny}">
    <value name="{název položky}" type="{typ položky}" default="{výchozí hodnota}"/>
    ..další elementy group nebo value
  </group>
  ...další elementy group nebo value
</info>

```

Složka /SyncFiles

Tato složka obsahuje soubory zajišťující synchronizaci. Všechny tyto soubory až na výjimku order.txt jsou ve formátu Xml, a obsahují elementy, které generují všechny třídy dědící od Managers.SyncEvents.Events pomocí metody GetXml.

Soubory s koncovkou .sync,

Xml reprezentace událostí

Událost přidání souboru:

```

<fileEvent timeStamp="{proměnná typu dateTime, kdy se událost vytvořila}" deviceId="{id zařízení,
z něhož soubor pochází}" diskId="{id disku z něhož soubor pochází}" path="{cesta z kořene disku
(začíná ,\)}"> <addFile category="{kategorie souboru}" infoFile="{cesta souboru s poznámkami
z HFOFolderu}"> </addFile></fileEvent>

```

Událost změny dat v poznámkách:

```
<infoChange timeStamp="{dateTime, kdy událost nastala}" path="{cesta k souboru({id disku}\\{cesta  
z kořene disku}}}"> <value group="{cesta skrz skupiny položek ({název skupiny}\\{název  
skupiny}\\...\\{název položky}}}"> <![CDATA[{data která nastaví položku na správnou hodnotu }]]>  
</value> </infoChange>
```

Soubor HomeFileOrganizer.exe.config

Zde jsou uložena nastavení pro aplikaci, kde důležité části jsou elementy settings:

- s name="broadcastPort", kde je číslo portu pro broadcast (viz níže)
- s name="HFOFolderName", je název složky, která se vytvoří v systémové složce ApplicationData, což je kořen pro ukládaná data (viz výše), dále je také je základem názvu souboru identifikující disk.
- s name="localDevId", je identifikátor zařízení, kde je k němu příslušný program spuštěn
- s name="UserToken", je uživatel programu, na všech zařízeních v síti, by měl být stejný

Sítové spojení (třída Managers.Connections)

Spojení programu, probíhá několika způsoby. A zajišťují ho 3 nekonečné úlohy (Broadcast, ListenBroadcast, ListenConnection). Tyto úlohy se spouští skrze metodu RunConnection, která vytvoří Task, který skončí po ukončení těchto úloh. Úlohy se ukončí nastavením CancellationTokenu shouldEnd skrze metodu SetClose.

Veškerá komunikace TCP komunikace je zastřešena třídou Classes.Communicator.

Broadcast(Connections.Broadcast)

Tato úloha vysílá (cca každé 2 sekundy) do sítě skrze UdpClienta jako broadcast zprávu:

```
„v1.MFO.MFF.works.bubyx;id={identifikátor  
zařízení, které tuto zprávu vysílá};ip={adresa  
na nichž toto zařízení poslouchá příchozí  
spojení (odděleny čárkou)};port={port na němž  
poslouchá příchozí spojení}“
```

Tato zpráva je vysílána na port definovaný v HomeFileOrganizer.exe.config (viz výše).

Poslouchání broadcastu

(Connections.ListenBroadcast)

Tato úloha poslouchá broadcast zprávy na portu uvedeném v HomeFileOrganizer.exe.config (viz výše), pokud přijde zpráva, která je ve tvaru

(„v1.MFO.MFF.works.bubyx;id={x};ip={y};port={z}“) a pokud {x} odpovídá identifikátoru zařízení v uživatelské síti, tak zkusí navázat spojení na všechny adresy v {y} na portu {x}, pomocí níže uvedeného protokolu. Před navázáním spojení ještě ověří, zda již se zařízením, které zprávu vysílalo, není navázána komunikace, nebo zda zprávu neodeslalo toto zařízení. Spojení se pokusí navázat v metodě Connectnion.TaskTryConnect. Pokud se podaří spojení navázat, tak se spustí úloha Communication.runCommunication s rolí žadatele (viz níže).

Poznámka:

Socket, který zprávu vysílá (resp. přijímá) má nastavenou možnost

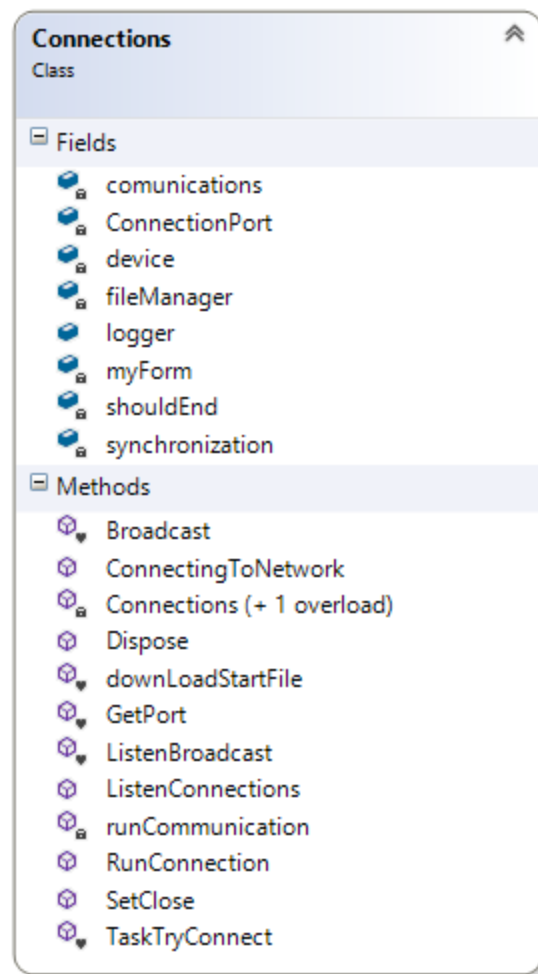
`SocketOptionName.ReuseAddress=true`, což umožňuje spustit aplikaci na stejném počítači, jinak by mohl mít systém problém se snahou přístupu dvakrát na stejný port.

Connections.ListenConnection

Tato úloha odpovídá na žádosti o připojení na systémem přiděleném portu (ten se nastaví do proměnné Connections.ConnectionPort) a odpovídá na ně (viz Protokol navázání spojení).

Žádosti jsou testovány, zda patří do sítě této instance aplikace. První podmínkou je, že zpráva obsahuje proměnnou user, která odpovídá lokálnímu nastavení

Properties.Setting.Default.UserToken. Tato proměnná odpovídá zahashovanému uživatelskému



jménu. Druhou podmínkou je, že id ve zprávě, pokud se jedná o hodnotu ulong, odpovídá id nějaké existující instance MyDevice, pokud ne odešle se zpráva o chybě. Pokud id ve zprávě není hodnota ulong, tak se id považuje za název nově přidávaného zařízení, tomu se vygeneruje identifikátor a započne jeho přidání do datové struktury.

Nakonec, pro každou žádost, která projde zmíněným procesem, se spustí úloha pro řízení komunikace `Connections.runCommunication` (viz níže) s rolí informátora.

Protokol navázání spojení

Zařízení snažící se navázat spojení

```
connect.v1.MF0.MFF.works.bubyx;id={identifikátor tohoto zařízení};user={řetězec  
specifikující uživatele}
```

`connection established`

-nebo-

`wrong informations`

`connection started`

nebo pokud se připojuje nové zařízení

```
connect.v1.MF0.MFF.works.bubyx;id={navez zarizeni zadnici u pripojeni};user={řetězec  
specifikující uživatele}
```

`connection established;{přiděleny identifikátor novému zarizeni}`

`connection started`

Komunikace (`Connections.runCommunication`)

Zajišťuje komunikaci a přenos dat, dle protokolu uvedeného níže.

Je vždy jen jedno spojení pro dvojici zařízení. Pro protokol může být zařízení v pozici informátora, nebo žadatele (viz níže), kdy informátor poskytuje informace o než si žadatel zažádá, tedy žadatel řídí komunikaci.

Práce žadatele

Žadatel pořád opakuje smyčku žádostí o další synchronizační soubor a zpracovává synchronizační události. Kdy je možnost komunikace předána instanci události, takže si může požádat o stažení souborů.

Žádost o synchronizační soubor

Žadatel

Informátor

```
getNextSyncFile;version={poslední verze o kterou si zažádal}
```

Pokud není nový synchronizační soubor

`fullySync`

Pokud existuje soubor pro synchronizaci

```
fileStart;{počet, kolik následujících bytů je soubor}
```

`{soubor}`

Žádost o libovolný soubor

Žadatel

Informátor

```
getFile;fromHF0={cesta ze složky s aplikačními daty}
```

Nebo

```
getFile;fromTree={cesta skrze aplikační datovou strukturu:{identifikátor  
zařízení}\\{identifikátor disku}\\{cesta z kořene disku}}
```

Za normálních okolností

```
fileStart;{počet, kolik následujících bytů je soubor}  
{soubor}
```

Pokud soubor nelze nalézt, či otevřít

```
fileError
```

Prohození rolí informátora a žadatele

Žadatel

Informátor

```
switchRole
```

Synchronizace

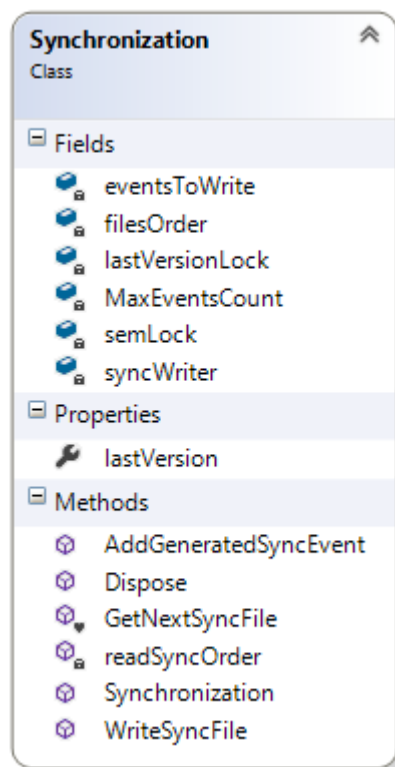
Synchronizace probíhá, tak že pokud aplikace na podnět uživatele dostane pokyn k nějaké úpravě dat, vytvoří jí odpovídající event a předá jej synchronizační části

(Synchronization.AddGeneratedSyncEvent). Synchronizační část když uzná za vhodné, tak vytvoří synchronizační soubor, kde jsou tyto eventy zapsány. Ten to soubor pak je poskytován zařízení, které žádají o synchronizaci.

Podporované eventy lze nalézt jako třídy v programu ve složce Managers/SyncEvents. Aby event fungoval, dědí od třídy Events, kdy musí pro svou funkčnost umět vypsát xml text pomocí GetXml, z kterého pak umí získat data pomocí svého konstruktoru s parametrem XmlReader. Dále musí umět upravit data aplikace pomocí metody DoEvent tak aby odpovídala změně, která tento event vytvořila.

WriteSyncFile

Synchronizační soubor se zapíše v případě ve dvou případech, pokud je přidána událost a fronta událostí (eventsToWrite) je větší než hodnota v MaxEventsCount, k tomu ještě musí být hodnota syncWriter=null. Druhý případ je, že někdo zavolá metodu GetNextSyncFile a neexistuje další soubor k synchronizaci.



semLock

Toto je zámek, který se má použít při přidání do události, nebo při úpravě položky eventsToWrite.

GetNextSynfFile

Tato metoda, zjistí jaký synchronizační soubor následuje po tom, jehož název je předán jako parametr. Je v parametru předán prázdný řetězec, tak je vrácen první soubor. Pokud není soubor k synchronizaci, pak vrátí null a k tomu spustí úlohu WriteSyncFile. Jinak vrátí soubor, který bezprostředně následuje po souboru v parametru.

Důležité

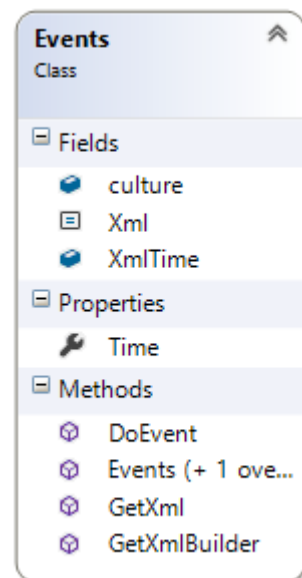
Je zakázáno spouštět více instancí WrtieSyncFile najednou, kdy právě běžící instance je v Synchronization.syncWriter, pokud nic neběží, pak je hodnota null.

Údálosti

V aplikaci je několik implementovaných událostí, kdy je důležité vědět, že všechny události by měli dědit od třídy Manager.SyncEvents.Events a reimplementovat její metody.

DoEvent

Tato metoda by měla zařídit, že data v počítači se upraví, tak aby odpovídali těm co jsou na zařízení, které tuto událost vyvolalo.



GetXml

Tato metoda vytvoří řetězec ve formátu xml, který je uložen do souboru. Může využít metody GetXmlBuilder události, od níž dědí. Která vrací ElementBuilder, pro snadné vytváření částí xml souboru.

Další konvence

Třída má pro všechny své (Fields a Properties) statickou proměnnou která definuje název atributu v xml souboru.

Třída má konstruktor, který dostane jako parametr XmlReader, případně nadřazenou třídu dědicí od events, který si přečte z xml souboru data. Xml element reprezentující událost má jméno určené statickou proměnnou Xml.

ElementBuilder

Tato třída zajišťuje rychlou tvorbu částí Xml souboru.

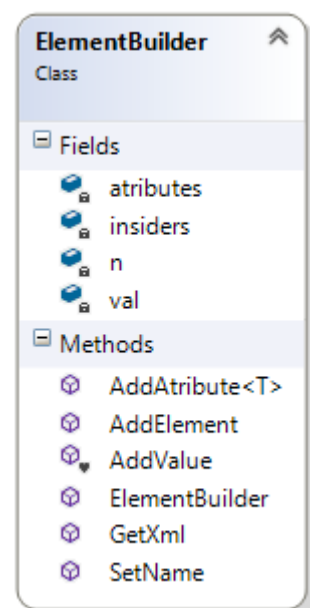
AddAttribute<T> přidá atribut s hodnotou k elementu

AddElement vloží do elementu jiný element

AddValue vloží do elementu hodnotu

SetName změní jméno elementu

GetXml vygeneruje řetězec se všemi vnořenými elementy, a tributů a hodnotou.



Možná rozšíření

Program je možné rozšířit jakoukoliv práci se soubory, kdy stačí implementovat nové sychronizační události, které je pak nutné ještě přidat ve čtení syncSouboru (XMLProcessor.XmlSyncFile.readSyncFile).

Program je částečně připraven na rozšíření správy celých složek souborů.

Program, má taky pár funkcí, kterým chybí uživatelské ovládání, jsou k nalezení ve složce Managers/syncEvents.

Nedostatky programu

Síťová komunikace není zabezpečená a může být napadena mnoha způsoby, tudíž by se hodilo doplnit síťovou část o šifrování a upravit protokol pro připojení. (Hlavní nebezpečí je v přidávání zařízení, kdy se může do sítě připojit v podstatě kdokoliv)

Uživatelská část

Pro uživatele, je připraveno základní ovládání v GUI, kdy pokročilý uživatelé mohou upravovat aplikační data a tím rozšířit své možnosti (v tomto případě je doporučeno si pročíst programátorskou část). V případě, že uživatel okno minimalizuje, pak program přesune z lišty programů do oznamovací oblasti.

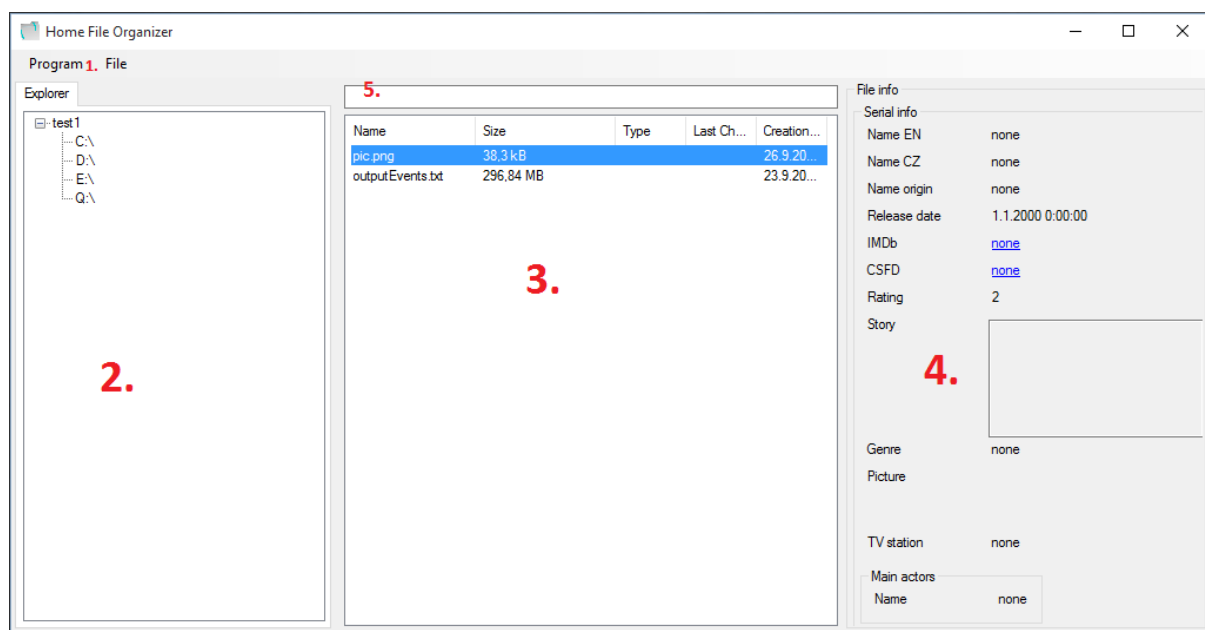
Synchronizace probíhá, tak že vždy zařízení na němž jsou vykonány úpravy předá o úpravě informace zařízení, které se k němu připojí. Z toho plyne, že úpravy se zobrazí na zařízeních, která od doby jejich provedení běžela souběžně se zařízením, kde byly úpravy vykonány (nešíří se informace pomocí přenosu mezi zařízením, ale jediné přímo z původce úprav). Tedy k úplné synchronizaci je třeba, aby všechna zařízení běžela chvíli souběžně s ostatními.

Při zavírání programu normálním způsobem se zobrazí okno informující o ukládání dat, v této chvíli pokud někdo program násilně ukončí, tak dojde pravděpodobně ke ztrátě dat a rozbourání datové struktury. Tedy rozhodně neprovádět.

Hlavní okno

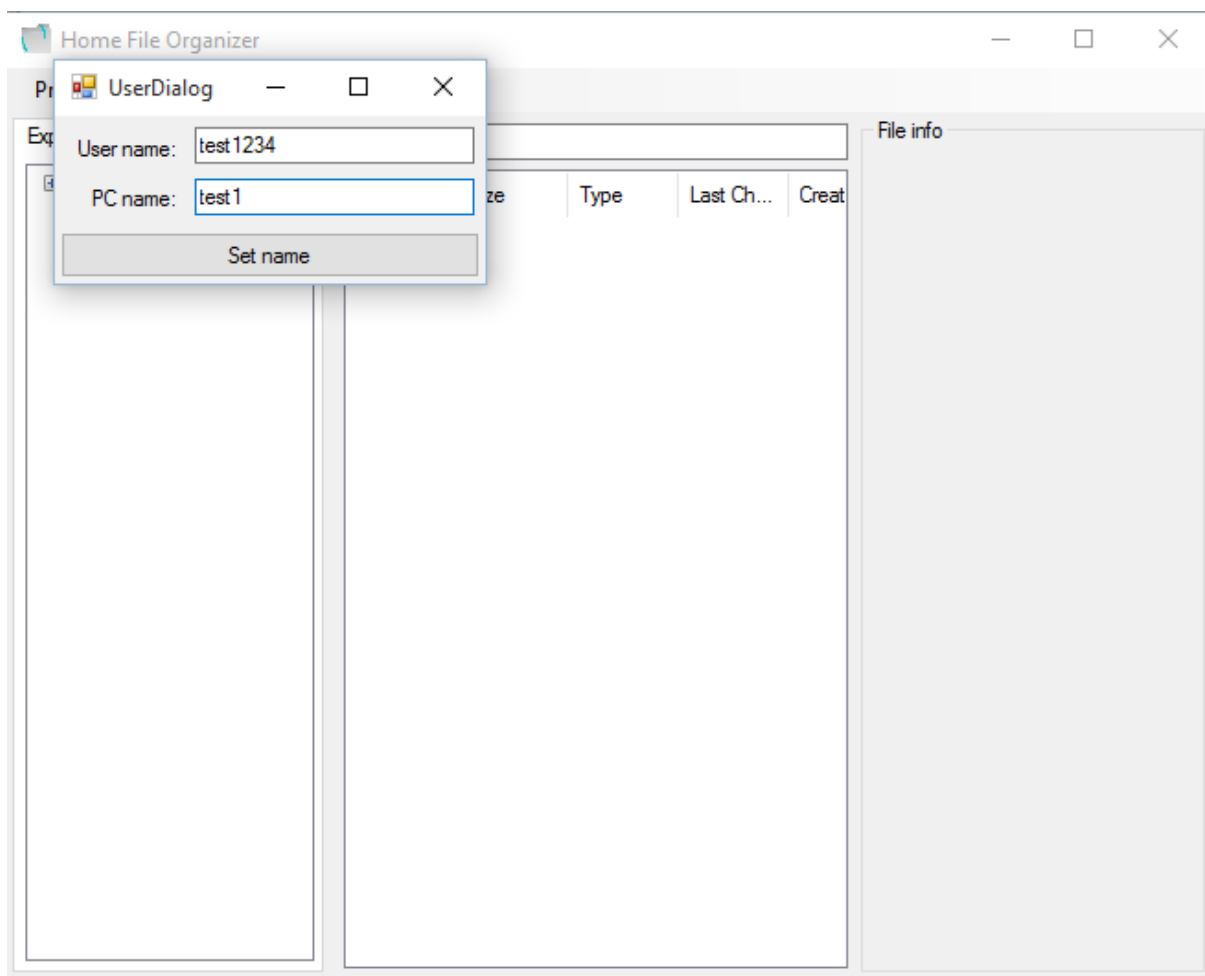
Grafické rozhraní, je okno které je rozděleno do 5 oblastí.

1. Menu, kde uživatel najde ovládací příkazy
2. Strom dat, zde uživatel volí, ve stromové struktuře zařízení, pak disk, případně složku
3. Seznam souborů, kde jsou zobrazeny jednotlivé soubory, které patří do části vybrané ve stromu dat, nebo soubory, které projdou hledáním
4. Informace o souboru, což jsou data které si uživatel připsal k souboru
5. Pole pro vyhledávání



Vytvoření sítě

Aby bylo možné program používat musí uživatel vytvořit programovou síť. To udělá tak, že spustí program s uživatelskými právy administrátora (viz nápověda systému), následně v menu vybere Program>Create new network. Pak se mu zobrazí okno s žádostí o uživatelské jméno (musí obsahovat minimálně 9 znaků a být bez mezer) a název zařízení.

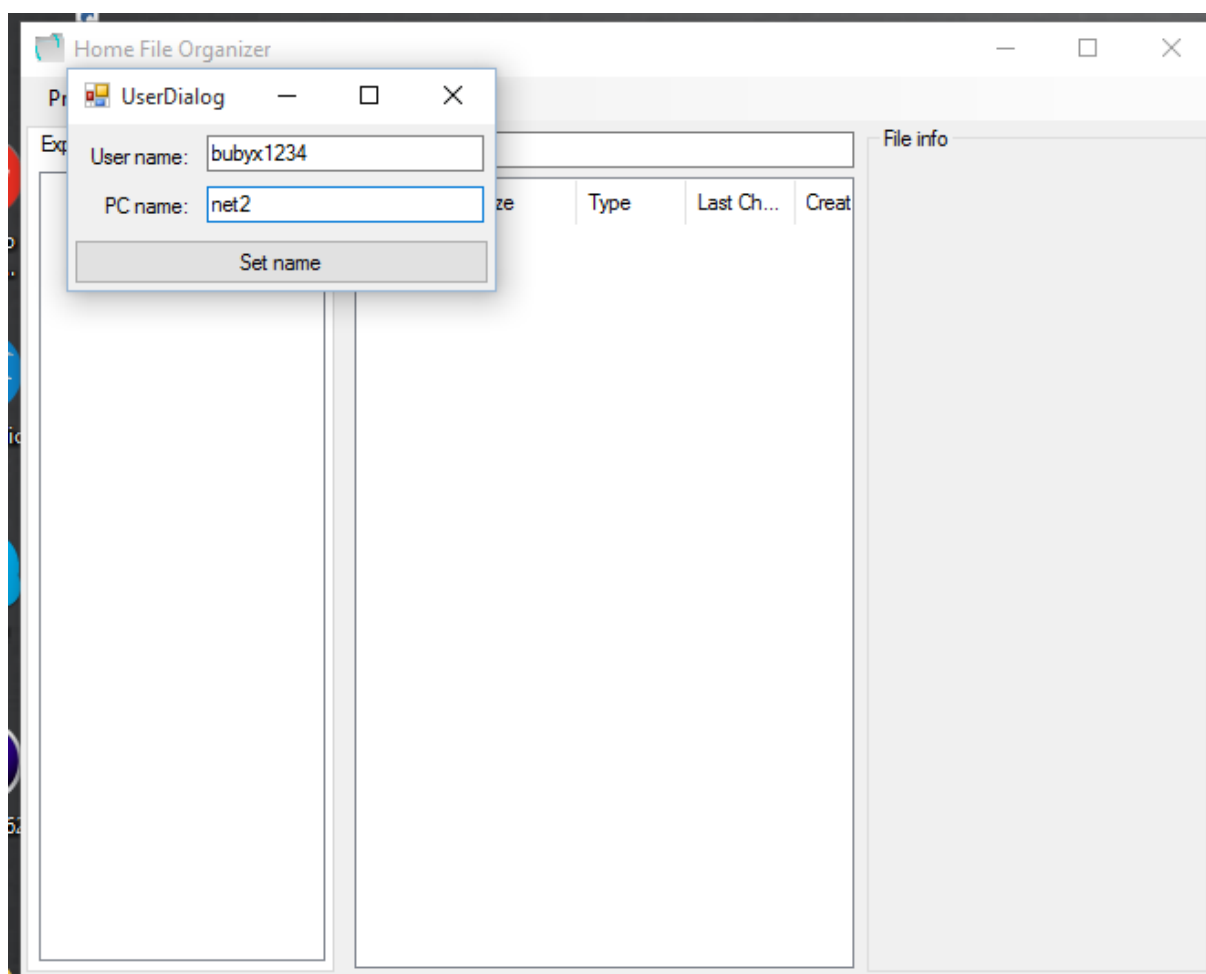


To vyplní podle své libosti. Je však nutné si uživatelské jméno dobře zapamatovat, neboť je nutné k připojení dalších zařízení.

Po vyplnění je třeba kliknout na tlačítko Set name, čímž se zahájí tvorba sítě. Následně musí uživatel počkat, než se všechna okna zavřou a znova se mu otevře okno, které již zobrazuje toto zařízení.

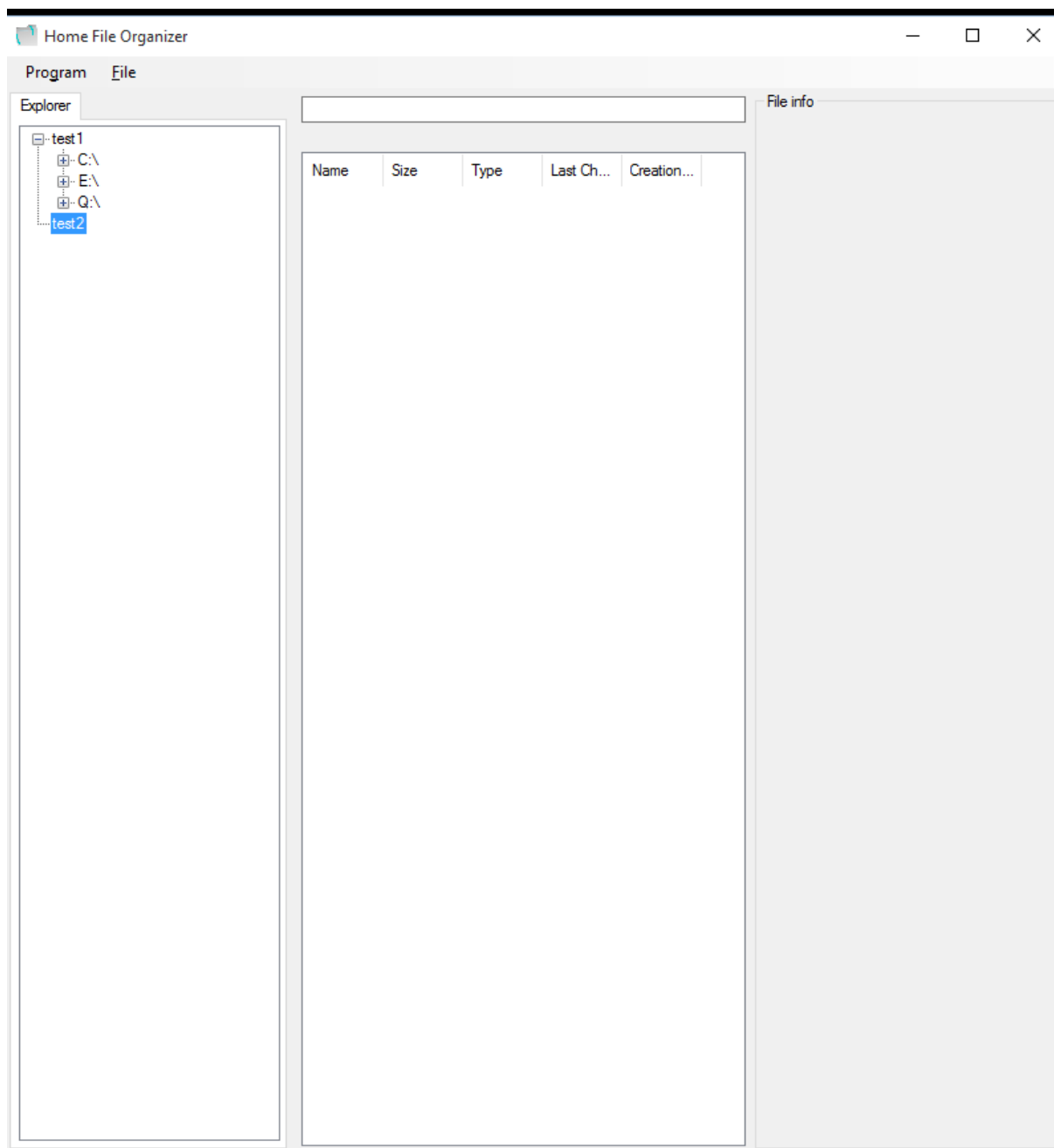
Připojení k síti

Pro připojení k síti, je třeba mít již vytvořenou síť a pamatovat si uživatelské jméno, kterým se do sítě přihlásíte. Pro přihlášení zvolte v menu Program>Connect to network. Následně se zobrazí okno pro vyplnění údajů. Do políčka User name zadejte uživatelské jméno, které jste použily při vytváření sítě.



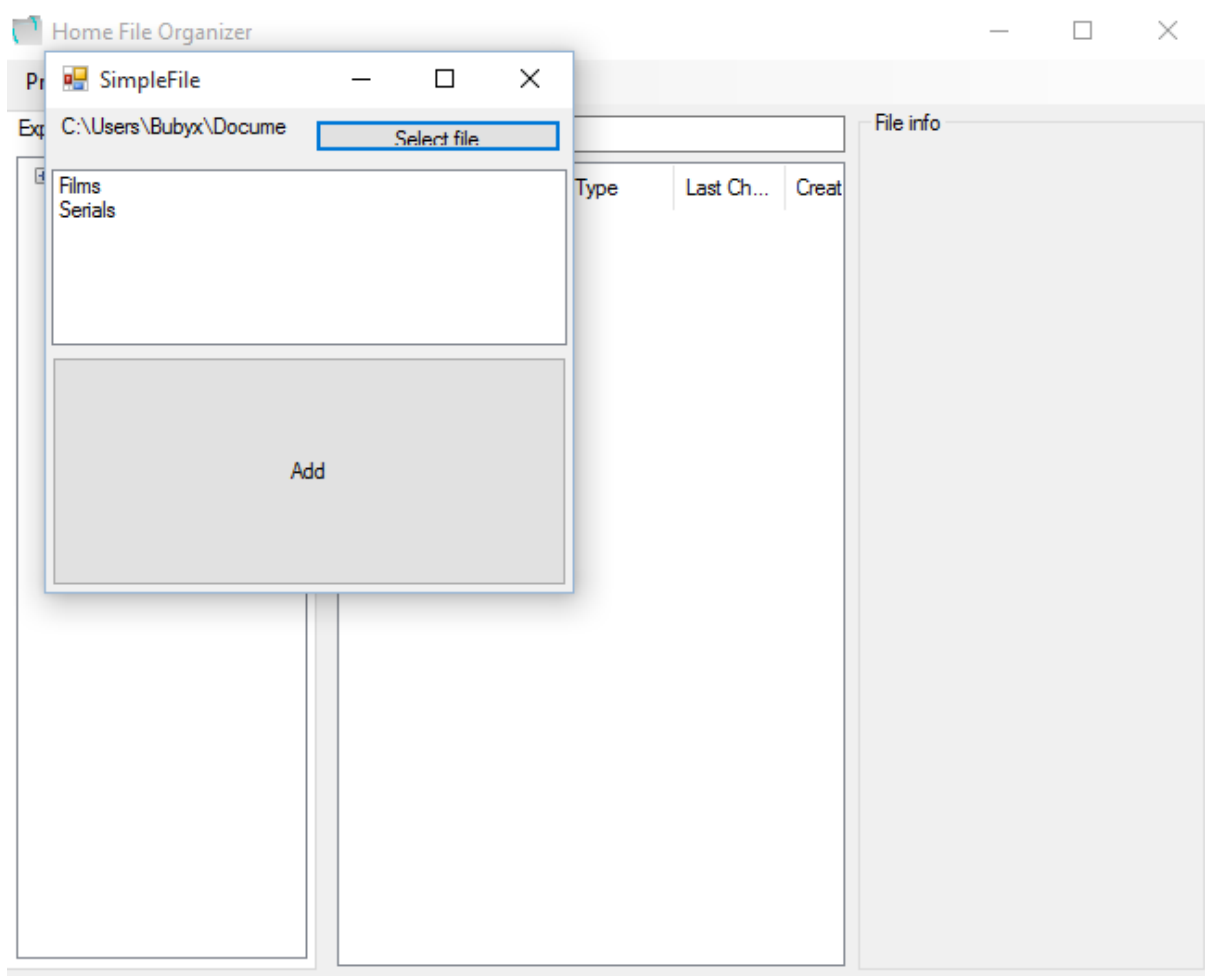
Po připojení, se okno párkrát zavře a otevře. Pak by vše mělo být připraveno k užívání.

Pokud náhodou nastane situace, že se na nějakém zařízení nezobrazují disky nově přidaného zařízení, jako na obrázku níže, tak postupně zapněte na všech zařízeních program a následně znova zapněte.



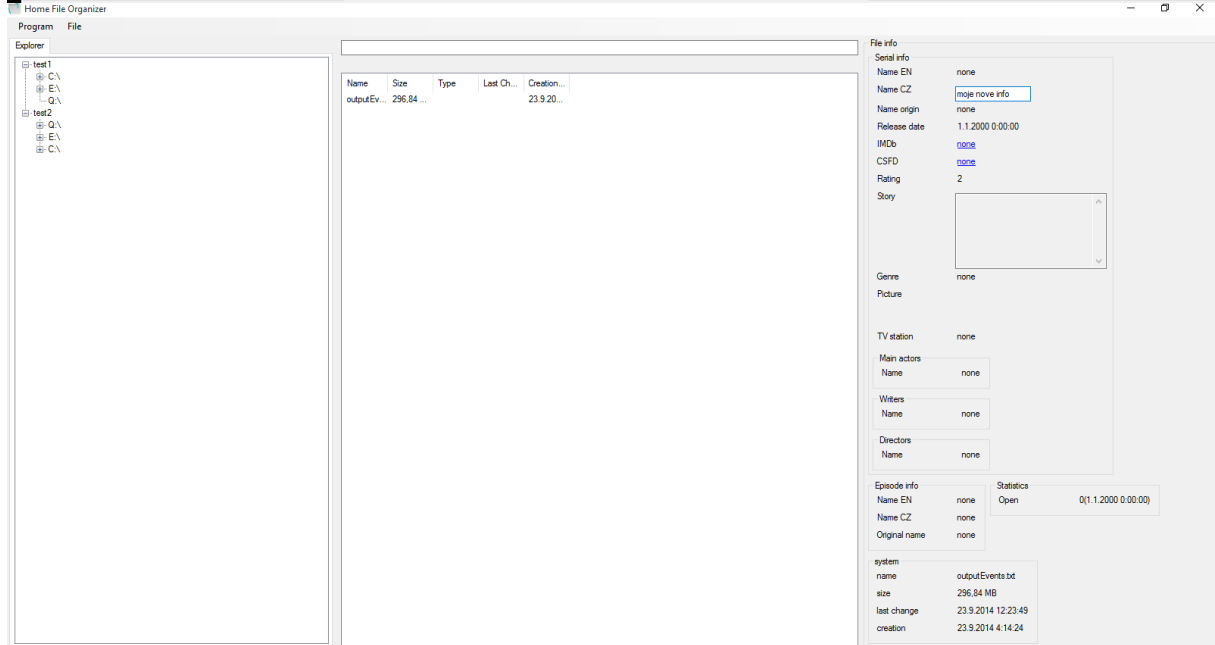
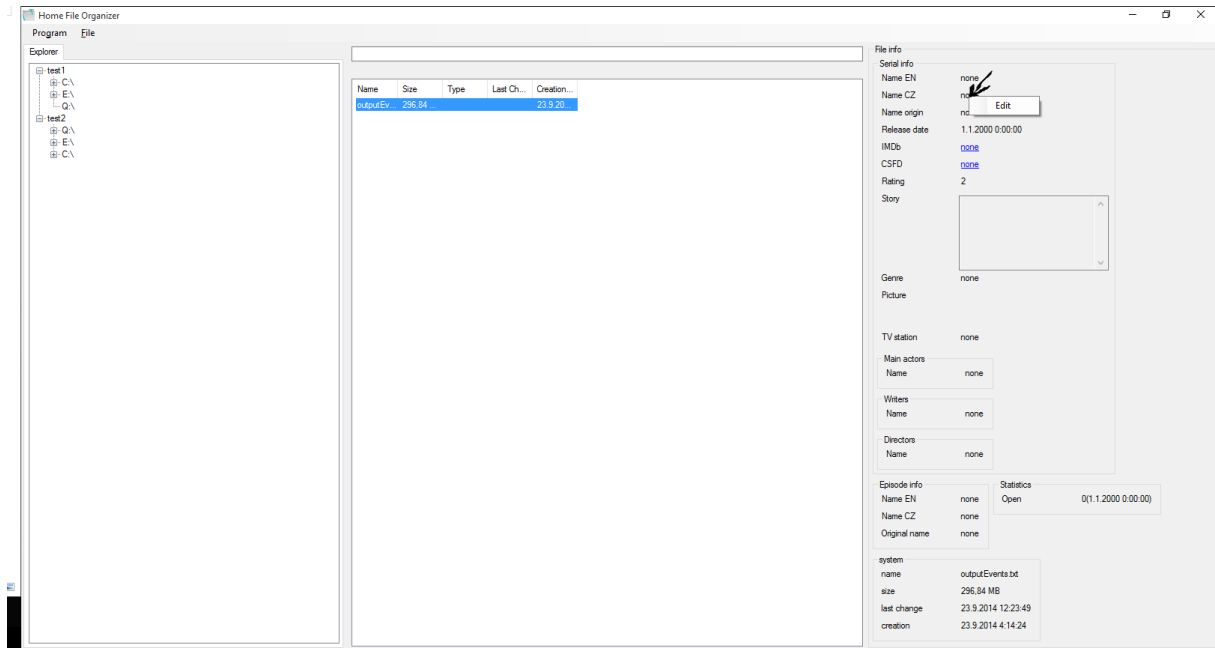
Přidání souboru s výběrem kategorie

Pro přidání souboru do programu uživatel vybere v menu File>Add>Simple file. Poté se mu zobrazí okno pro výběr souboru a kategorie. Kdy může vybrat vždy jen jeden soubor a jednu kategorii. Okno mu také zobrazuje cestu vybraného souboru. Po vložení validních dat uživatel klikne na tlačítko add, které zahájí potřebnou práci programu. Pro rychlé přidání Filmu (resp. Seriálu) je v menu položka File>Add>Film file (resp. Serial file), které zobrazí pouze okno pro výběr souboru.



Editace položek

Pro editaci položek, které se zobrazují u souboru, je nutné kliknout pravým tlačítkem myši na hodnotu položky, následně se zobrazí možnost Edit, tu zvolte. Tím ze zobrazovače hodnoty vznikne upravitelné pole. Zároveň lze editovat několik položek. Položky pak uložíte pomocí Program >Save infos, avšak editovací pole zůstanou k dispozici dokud nepřejdete k jinému souboru.



Vyhledávání

Vyhledávání probíhá na základě uživatelských poznámek, k souborům. Kdynelze upravovat prioritu operací pomocí závorek a jsou podporovány jen operace „logické a“, „nogické nebo“ a porovnání ($<$, $>$, $=$, $<>$, $<=$, $>=$).

Hledání tedy probíhá, tak že je zadán cíl, způsob porovnání s hodnotou a hodnota. Více požadavků, pak lze spojit pomocí spojek | a &, kdy & váže více než |.

Cíl je definován stromem poznámek, tedy názvy skupin (názvy vepsané do rámečku, avšak počáteční rámeček File Inko, sem nepatří) oddělené znakem \ a na konci název položky. Pozor nesmí nikde přebývat mezery.

Pr.: „Film info\Name CZ=Králova řeč|Film Inko\Name EN=King's speech“

Další ovládání v menu

Další nástroje programu jsou v menu File>Transfer

Kde File>Transfer>Move přesune souboru z jednoho zařízení na zvolené zařízení a File>Transfer>Create copy vytvoří kopii souboru na zvoleném zařízení.

Postup pro oboje je následující.

V seznamu souborů se vybere soubor s nímž se bude manipulovat (bude modře obarven).

Pak v menu vyberte File>Transfer>Create copy (resp. Move). Tím se zobrazí nabídka se zadními, kam se má soubor zkopírovat (resp. Přesunout). V nabídce je zobrazen vždy název zařízení a v závorkách jeho id.