A little step to a Smart House: Relational File Systems

Research Project Description

Roberto Valenti

Topic:

Looking to the coming years, it's possible to imagine smart houses that control the temperature of the water while having a shower, organize the personal data, synchronize with the mobile devices, controlling the energy consumption, managing all the little smart devices that it's possible to imagine.
The main problem in the state of the art of computer science is that the calculators cannot understand the data itself. In the current filesystem model, data is stored as bytes, and the file is just a container. All the remaining information is attached by the user and usually consists only on a name and a file extension to understand the type of data, while the directories are just a way to hierarchically self-organize the contents. This environment is not helping the development of such a revolutionary vision.

Research Hypothesis:

We need to redefine a modern storage concept featuring a combination of the file system services and the relational system to provide a new powerful and enhancing storage platform. The purpose is to elevate the file system conception to a higher level where data interoperation and global data interchange will be possible.

Approach to reach this goal:

The basic idea is to create a filesystem with predefined data-types (templates). The files will be stored in a relational way described by the correspondent data-type schema. Depending on the data, the relations can be strong or weak. The strong relation can be compared with the "old" directory structure with some extensions (I.e.: An album is a strong related knot of songs, an artist is a strong related knot of albums. If an album or an artist is deleted, all the songs from the album / all the albums from the artist will be deleted as well) while the weak relation is a relation that can be broken without influencing the rest (I.e.: If we want to unlink the "friendship" relation from a contact to another, we don't expect to delete any of the contacts, since they can be related to others). Note that the weak relations cannot modify the fields of the stored file, but just how that file is weakly linked to the others. If a file is not weak-linked to and from any data-type, the file will be deleted. Furthermore, the weak relation can point outside the main system: in this way the system knows that there is a file on a mobile device, and can find it even if the mobile device is not connected to the system. This can lead to very nice enhancements when the mobile device is on another smart-house environment (like an office), that can be used as a link between the two systems, exchanging data and relations between the two environments. The same approach can be applied to all the data-type (Music, videos, images, e-mail, text documents etc.).
The physical files will be stored all together on the disk, with the preference of having the same data-type in the same track on the disk to speed up the researches (no needs to move the heads while searching/reading files, unless multitasking is needed) and can appears to the user in a user-friendly way: it's possible to create a directory listing, where every directory is actually a query to the filesystem. (I.e. we can reach the same physical file of a song by browsing the artist directory, the year

directory, the genre directory or even the mapped directory of MP3 player history to understand when the song was last played.)

The use of xml is suggested/necessary to keep compatibility between all the electronic/smart devices (Mobile, PDA, TV, heating system etc...). In the moment that a new peripheral can access the main system (by internet, wireless, or simply docking the peripheral etc..) It would send an xml files with the files and relations modified since the last "docking", and the main system will decide which files to request or to send to the peripheral to synchronize, building up the new relations. This project can be extended with a lot of nice features, but those will not be discussed in this brief description.

Time needed:

The basic form of the project can be done in around one year, with a consistent group of database and filesystems experts. The extensibility of the innovations that can be developed using this approach can require decades and the integrations of other producer companies, to reach the utopian dream of a smart house.

Risks:

The risk of this project is mainly about its performances. While there are almost no limits to the possible uses of this architecture, there is a performance limit that depends on the speed, the memory of the system and on the kind of requests. Keeping this over fitted network of relations between files may bring the system to a recursive loops during updates or searches. Those last problems can be solved with the use of ad-hoc algorithms or depth-limiting the operations, but the performances can be affected by a lot of other problems.

Furthermore this project needs to be accepted by the users and by the hardware producers to be fully operative.

The xml is a good and extensible standard that will limit the risk of a failure due the use of a proprietary communication protocol that will be surely rejected from the user and producers community. Anyway the risk of failure, due to the big purpose of this project, is pretty high.