

Recap

Today, the life of people is dominated by computers and mobile devices, however dominant era of conventional PC's and Smartphones are coming to an end because of number other web service enable devices are growing rapidly. Use of multiple devices becoming a common matter, but as a user we often face problem of device management, for example some device doesn't support some platform/software, inter connection among devices, data backup / synchronising etc. To get rid of this problem new concept is undergoing is called "Liquid Software" an approach in which applications and data can flow from one device or screen to another seamlessly, allowing the users to roam freely from one device to another, no longer worrying about device management, not having their favourite application or data, or having to remember complex steps ¹. Software have been playing a vital role since IT industry grew up, to enable its liquidity characteristic software will be built dynamically combining the best available components for each purpose by downloading them dynamically from different websites.

Nowadays an average consumer in Europe or U.S has two computing device (usually laptop and Smartphone), trend of having third one (Tab) is growing. Everyday 3.5 million new devices are activated worldwide, may be within short time we shall move to an age where each of us will have dozens of devices for daily use. Number of developers are actively looking into alternative target devices (e-readers, TVs, set-top boxes and game consoles) for future software development efforts. The trends toward multiple device ownership would aggregate the demand of software liquidity.

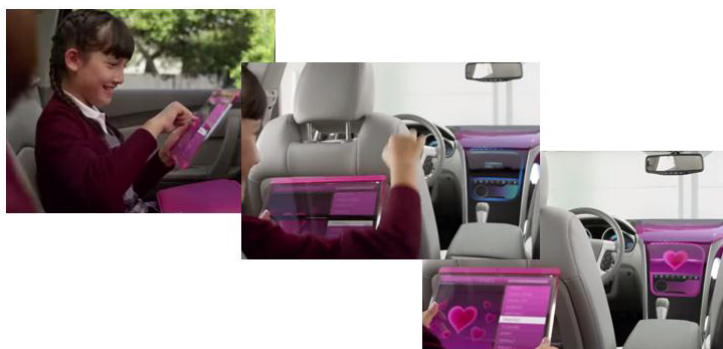


Fig: Illustration of liquid software (tab to car)

user interactively transferring and synchronizing the music player app and its theme from her tablet to the car's audio system. ²

The manifesto of liquid software are

- Users shall be able to effortlessly roam between all computing devices.
- Roaming would be casual, fluid and hassle-free
- Applications and data shall be synchronized transparently
- Roaming shall not be limited to devices from a single vendor only.
- The user shall remain in full control regarding the liquidity of applications and data.
- For ensuring seamless continuation of activity on any device roaming shall include the synchronization of the full state of each application

Problem

It is a very new topic to me. I guess it is over all imaginary view how the characteristic of liquid software should be. Haven't got way to implement it. Data / app transformation between homogeneous devices (mobile, tab, pc) was stated but how heterogeneous devices (washing machine, air condition, microwave oven) can also take part would have been nice to know.

Criticism

From the very notion of the user's join in liquid software environment most of the applications and data being accessible from any of the user's devices can make the system vulnerable from the privacy perspective. For instance, entire computing environment can fall in security threat even if a single device of the user is stolen. User control can be the simplest way to ensure security in this case; liquid software may get access by setting up access control lists for the resources such solutions must not be made too intrusive from the user's viewpoint.

Deepening

As liquid software has to run on a heterogeneous collection of machines therefore compiler in different machines should be able to compile the code to avoid corruption at execution state. So the goal is to compile the software very fast while it is transferred over the network. As initial step Sun Microsystems developed Java bytecode for machine-independent code representation, the machine-independence achieved by fixing they byte-ordering, data alignment, data layout and so on to meet standard. The bytecode is based on a small stack-oriented instruction set that is easy to interpret and should be simple to translate to native code.

Reference

1. <http://lively.cs.tut.fi/publications/LiquidSoftwareManifesto-2013.pdf>
2. <https://www.youtube.com/watch?v=jZkHpNnXLBO>

3. ftp://158.130.67.137/pub/cis700/public_html/papers/Hartman96.pdf

Khandker, Syed Ibrahim

WISE