MPL licence is like LGPL + an exception to be able to link statically with proprietary software + no patent grant to you or your users if you use a modified version of the software. If you modify the MPL covered code, you must provide the modified MPL source in a working form with all additions. The final user can ask the distributor for the MPL licensed source code.

A module covered by the GPL and a module covered by the MPL cannot legally be linked together.

Cairo source can also be under LGPL. U++ is available with its source code so no problem but:
- With static linking, if someone else use your software + cairo under LGPL, the final user can ask the distributor for the cairo source code (modified or not) + your source. Your source must be LGPL or GPL. If the final distributor link statically to your software, he must release the wall source using GPL or LGPL. With dynamic linking, the final distributor will just have to add in its docs, apps, about menu, ... that his software use LGPL licensed code and must provide the source of the LGPLed code with its software (in the media!).
- If they link statically with U++ and U++ link dynamically with the cairo LGPL covered source, the distributor can use a proprietary licence for their own software but still need to tell about the LGPL licence of the dynamic library.

Quote:Great, you have to be a lawyer to make heads or tails out of this licensing issue.

Sure!

Most of the FOSS licences let you include other source code covered by the same licence into your own source. When you select an open source licence, you should carefully study the advantages, drawbacks and risks. With a not enough spread one, you will have to reinvent the wheel each time you want a new feature. The more licence we have, the more fragmented the community is, the more difficult producing new FOSS become.

Really, before any work on including other libraries, can we have a definitive statement about the final U++ licence choice?