Mag. iur. Dr. techn. Michael Sonntag

Legal protection of component metadata and APIs

17th EMCSR - 2004, Vienna, 14.4.2004

Institute for Information Processing and
Microprocessor Technology (FIM)
Johannes Kepler University Linz, Austria

E-Mail: sonntag@fim.uni-linz.ac.at
http://www.fim.uni-linz.ac.at/staff/sonntag.htm

Introduction

- Metadata = Data associated with other data ("main data")
  - → To explain, support, or describe the main data
  - → But not directly required for the functioning of the main data
- Several kinds of legal protection can apply to metadata:
  - → Copyright protection: The individual expression is protected
      - – E.g. the written description of a new kind of screw
  - → Patent protection: Can also protect the idea/method behind the expression
      - – E.g. the screw's characteristics, regardless of method of description
    - » Problem of software: Description and idea merge easily
      - – An algorithm is an idea and its description at once
      - – Is the problem patented or the solution?
    - » Rarely applicable to metadata instances!
  - → Sui generis protection: E.g. Metadata for rights protection
  - → Trademarks, servicemarks, etc.

Metadata and components

- → Documentation: Important for understanding: See copyright!
- → Code information: Runtime information for dynamic calling or assembly of code
  - » Created automatically; might be a problem for protection
- → Environmental or commercial data: requirements, support offers, weblinks to additional information, ...
  - » Usually too small for copyright to apply
- → Prerequisites, incompatibilities: For assembling components
    - – See also configuration files (e.g. which modules to load)
  - » Must sometimes be changed/added by aggregator!
- → Orthogonal requirements: Synchronisation, isolation, transactions, reentrance capabilities, etc.
- → Licenses, rights information, DRM data: Textual and/or structured information
  - » Copyright (+ special!) protection

General copyright issues

- For metadata to be protected, it must be a "work"
  - → From a protected type: Usually "literature" here
  - → Human originality: Must be "created", not "discovered"
  - → Certain level of originality: Generally very low
    - » But might be a problem for individual pieces of metadata!
- Longer written documentation: Protected, no problem
  - → Any changes (modification, integration into other works; also for parts) require the permission of the copyright holder
  - → Facts contained are NOT protected!
    - » Taking the interface and creating new documentation is allowed
- Most metadata will however be rather "computer programs"
  - → Program (=Literature) encompasses all forms of the code and all the materials for development (e.g. diagrams)
    - » Less protection! E.g. changes allowed in certain circumstances

General copyright issues

- Fully automatic created data / data derived from code:
  - → These itself are facts, but derived from "original" code
  - → Therefore they are indirectly also original and protected
- Special problem of metadata: Small size
  - → Several gradual levels:
    - » Whole program, documentation, ...: Obviously a work
    - » Method, paragraph, ...: Depends on size and creativity
      - – Is it a recognizable part of the whole or could it appear anywhere?
    - » Individual instruction, word: Too small
      - – But several together are perhaps protected as a collection!
        - » Is a program **more** than a collection of statements (Holism vs. Reductionism)?
  - → Metadata is mostly on the brink
      - – And therefore legally "difficult"!
    - » Individually unprotected
    - » Protected as a whole or in combination with the software itself

Copyright protection of APIs
Introduction

- Most components provide more complex services, requiring an elaborate interface: An API
  - → Several methods
  - → Calling restricted (order, circumstances, callers, ...)
- An API is usually the result of a complex design process
  - → Functionality: What will the individual methods do?
  - → Naming: How will they best be called and documented?
  - → Grouping: Which pertain together, which are related, ...?
- Legal protection:
  - → Individual methods probably not protected: Too few creativity contained and too few other possibilities
  - → API as a whole usually protected: Selection of methods
    - » Requires more than e.g. images, as no arrangement present
      - – If objectoriented: Classes could perhaps be seen as "arrangement"!

Copyright protection of APIs
International situation

- USA: Some relevant judgements to the contrary
  - » But not directly on APIs (e.g. user interfaces, file formats)!
  - → "Elements dictated by external considerations can never be copyrighted". But:
    - » An API is not dictated by existing external reasons, only its counterpart (programs using the API) are dictated by the API!
    - » APIs are usually not just an "image" of the real world, but involve creativity and selection between different options
- EU: "proprietary" APIs are explicitly acknowledged
  - → Encourages providing information for implementing them
    - » Electronic Communications Framework Directive, Art. 18 para 2
  - → Computer program copyright directive:
    - » "...ideas...which **underlie its interfaces**..." cannot be protected
    - » The interfaces themselves could therefore be protected!

Limits of copyright protection of APIs

- However, even though an API is protected, this protection is far weaker than other copyrights ...
- Special provisions for computer programs exist:
  - » This applies only to the interface, not to the code!
  - → Right to make adaptations and copies for the intended use
    - – The commercially most important part!
  - » Creating "copies" in a program using the API is allowed
  - » Once a copy exists ⇒ "original" API can also be reimplemented!
- Derivative works (=modifications of the API) still require permission of the owner
  - → Could be a tool against incompatible modifications of the API
  - → Difficulty: What is a modification and what is "grouping" with an independent other API (which IS allowed)?
    - » A bit easier to decide with objectorientation: Deriving a class!

Copyright protection of APIs Reimplementation

- But: No requirement to provide information about the API:
  - → Reimplementation allowed, but actually impossible?
  - → Decompilation is allowed if necessary to obtain information for adaptation to another program
    - » Retrieving the API, its exact specification, or its associated metadata (e.g. transactions, sequence)
  - → This information may not be used for anything else
    - » E.g. directly as documentation for the own implementation!
  - → Creation of a substantially similar software prohibited
    - » Only for creating client programs, not for reimplementation
      - – Way out: Document the client nicely and use this documentation for implementation of a counterpart (which is similar to the original!)

APIs can be protected by copyright, but the commercially important uses are allowed (but not necessarily easy)

Metadata and searching

- Important aspect of components is discovery
  - » Or there will be no reuse!
  - → Direct searching for components vs. searching for webpages about components
    - » Direct searching requires metadata
      - – Fulltext search probably useless
    - » Searching for webpages also requires metadata for good results
- Similar to metatags in webpages
  - → Legitimate reason for inclusion of "foreign" metadata
    - » E.g. competitors trademarks, product names
  - → Currently rather unclear: Completely unrelated keywords
    - » No customer diversion? User's expectations? Unfair competition?
  - → "General" metadata (OS, libraries, etc.) can be used freely
      - – As long as it is not misleading!

Protection of rights information

- Metadata on rights is specially protected:
- Protected is (regardless of form: text or encoded):
  - » Data provided by the rights holder: Not, if added by others!
  - » Identifying the work (e.g. title), author/any other right holder
  - » Information about the terms and conditions of use
- Against:
  - » Knowingly: No intention for defrauding needed!
  - » Modifications: Alterations, deletions (additions?)
  - » Distribution, importing, making available, ... of works, where such information has been modified
  - » Without authorization
- If the person knows or has reasonable grounds to know, that
  - » this induces, enables, facilitates or conceals
  - » an infringement of any copyright or related right

Protection of rights information: Application to components

- Metadata unprotected by this:
  - → Supported OS, functionality, configuration, incompatibilities, watermarks, decryption keys (information on other data!),...
- Specially protected is:
  - → Name of the component, creator (author/company)
  - → License agreement (little importance; protected anyway)
  - → Hardcoded number of licenses, program code for key verification (?), time limit, decryption keys (for this work), etc.
- This applies to source code as well as compiled code!
  - → Removing previous author's name from source is not allowed
    - » Sometimes possible: Company owns rights, and authorized by it
- Might be a problem with integration of several components
  - → Information must be removed at the original component and instead added to an outer wrapper: Facilitating?

Conclusions

- Component metadata can be protected by copyright
  - → Special problem of small size: Sometimes collectively protected
  - → See also trademarks and servicemarks
- APIs can be copyrighted, but the most important uses are free (including reimplementation)
  - → Information required can not always be obtained
  - → Copyright could be used against incompatible modifications
    - » Difficulty: incompatible implementation vs. modification!
- Metadata intended for indexing is similar to metatags
  - → "Foreign" data is partly prohibited, partly no problem
- Some metadata is more equal than others
  - → Name, rights holder and license are specially protected
  - → Great care must be taken on integration (or contracts)

Questions?

Thank you for your attention!