

COM6655 Professional Issues

Autumn 2021-22

Contract Law (part 2)

Professor Guy Brown

Department of Computer Science, University of Sheffield
g.j.brown@sheffield.ac.uk

Mid-term test

- Released 9am Monday 1st November
- Submit by 9am Tuesday 2nd November
- Administered via Blackboard (link will become visible when the test is released)
- Not a timed test, but should take about 30 minutes
- 8 questions covering topics to this week
- Must be completed in a single session
- Accounts for 20% of your module mark

COM6655 PROFESSIONAL ISSUES

SLIDE 2

Aims of this lecture

- To explain contracts for bespoke (custom-written) software.
- To explain legal issues associated with other forms of software contract (shrink-wrap and click-wrap licenses).
- To explain legal issues around contracts between software authors and publishers.

COM6655 PROFESSIONAL ISSUES

SLIDE 3

Contracts for bespoke software

COM6655 PROFESSIONAL ISSUES

SLIDE 4

Bespoke software

- Appropriate software may not be available 'off-the-shelf' for certain specific tasks.
- Software can be written or adapted by a specialist software firm — a 'software house'.

Common sections in contracts for software:

- definitions
- licence agreement conditions
- contract price and payment terms
- specification of the software
- time for completion, and terms for early or late completion
- arrangements for maintenance and enhancements
- escrow
- intellectual property rights

Definitions

- The first clauses define:
 - The parties to the contract;
 - Hardware on which the software will be installed; and
 - Other software with which it must operate (e.g. the operating system).
- To save space, and assist standardisation:
 - Client's full business name will be abbreviated to a term such as 'client' or 'customer',
 - Developer's full business name will be abbreviated to a term such as 'developer'.
 - The terms 'software' and 'hardware' will be defined the specific system to be used.
- This assists with the readability and interpretation of the contract.

License agreement

- Usually, the client will **licence** the software rather than have ownership.
 - A contract for writing software is therefore — in most cases — a licence agreement.
- If it is especially important for the client that the software it requires is not sold elsewhere, it should insist on an **exclusive licence** or an **assignment of copyright**.
- **Q. Will an exclusive licence cost more or less than nonexclusive licence?**
- **Q. Why might a client insist on an exclusive licence?**

Licence: duration, transferral, scope

- **Duration**
 - A licence must be for a fixed period of time.
 - If a duration is not stated, it is likely that the licence will endure for as long as copyright subsists in the software.
- **Transferral**
 - The contract should state whether the licence can be transferred to a third party.
 - By default, licence agreements are usually assumed to be transferable.
- **Scope**
 - Is it permissible to run the software on more than one machine?
 - If the buyer is a member of a group of companies, can the other members of the group also use the software?
 - Can the software be transferred from one member of the group to another?

Contract price and payment terms

- A contract is usually in the form of a licence agreement, so payment is usually termed a **licence fee**.
- However, the fee may be called the **price** since it may also include training and documentation.
- The price should be stated precisely, but:
 - Contract should also provide a mechanism for calculating the cost of additional work outside of the terms of the contract;
 - Easiest way is to state an hourly rate for programmers, analysts etc.
- If the price is a lump sum, does it include: maintenance and training? documentation? the cost of storage media?

Ambiguity with payment

- The contract should ensure that there is no ambiguity regarding when payment should be made, including:
 - If the client is late paying, will they be charged interest?
 - What if the client refuses to pay?
- **Q. Under what circumstances might a software house refuse to fix a price for a contract?**

Specification of the software

- The specification of the software should indicate the following:
 - A detailed description of the tasks the software will perform.
 - The equipment on which the software will run.
 - How quickly the software will carry out the operations required, bearing in mind any requirements for networking and concurrent use.

Ambiguity with the specification

- Often the client isn't sure what they want, so:
 - They may need to change the specification during the development process.
 - The original contract must allow for these changes.
 - If the changes to the specification are considerable, it may be best to terminate the existing contract and negotiate a new one – **novation**.
- **Q. What approach might you take if the client is vague about the specification?**

Comyn Ching Ltd v Radius plc (1997)

- Comyn Ching Ltd (CCL) wanted to integrate its computer systems.
- Radius plc was appointed to do this.
- Radius plc twice offered to carry out a feasibility study before the contract was signed, for which the cost would have been £6930. However, CCL:
 - Turned down this offer on both occasions.
 - Had little knowledge of computers, but decided not to employ a consultant.
 - Were not satisfied with Radius plc's solution, and sued them for >£3m damages.
 - Argued that Radius owed them a **duty of care**, and that this included investigating their requirements beforehand at no additional cost.
- Verdict?
 - The judge rejected this – he considered CCL's requirements to be a 'moveable feast'.

Time for completion

- The usual method of dealing with late completion is to include a term in the contract which gives the client a right to **liquidated damages**.
 - These may be quantified as a certain sum of money for every week completion is late (e.g. £200 per week).
 - A sum for liquidated damages must be a genuine estimate of financial losses that the client will suffer as a result of late delivery, not a penalty.
- When does the client know that completion has taken place?
 - Concept of **substantial completion**: a large percentage of the agreed price is paid on the completion of a substantial part of the system, and the rest is retained until the remaining work has been completed.
- The meaning of 'substantial' should be defined in the contract.

Maintenance and enhancements

- Many bugs appear only after a long period of time.
- If a bug appears this will be a breach of warranty, and the client can ask the software house to correct the error.
- The software house will wish to limit its responsibility for correcting such errors to a specified period of time.
 - A software house will usually offer an ancillary contract for maintenance.
 - This agreement will also provide for enhancements/updates of the software.
- Contract should state whether client wants to modify software themselves:
 - Recall the rules about correcting errors in the Copyright (Computer Programs) Regulations 1992

Escrow

- What happens if a software house goes out of business?
 - Will its clients be able to maintain and modify their software or find another company to do this for them?
- Many contracts include an **escrow clause** to cover such situations.
- Escrow is a form of insurance or guarantee, should something happen to the software house:
 - The software house deposits material with an independent person, the source code of a system, plus copies of all the design documents (e.g. UML diagrams).
 - If the software house goes out of business, then the source code and documents will be released to the client, who will have everything needed to arrange support for the software.

Intellectual property rights

- The contract may impose duties on both parties in regard of intellectual property rights.
- **Q. What rights will the client want?**
- **Q. What rights will the software house want?**
- As well as contract terms, copyright law and the law of confidence will give protection to both parties.
- The client should insist on an **indemnity term** in the contract
 - Prevents legal action being taken against them, if the software infringes the intellectual property rights of a third party.

Other terms and standard contracts

- A contract for writing software may also need to cover:
 - Training of the client's staff,
 - Conditions for termination of the licence,
 - and others ...
- In many cases, it will be possible to use/adapt a standard contract.
 - Most software development houses will have their own standards.
 - Commercial organisations can supply templates.
 - Professional organisations can provide advice:
Chartered Institute of Purchasing and Supply <http://www.cips.org>.

Independent professional supervision

- It may be advisable to have large contracts overseen by an independent professional consultant.
- The consultant would be responsible for:
 - ensuring that the specification is met;
 - general supervision;
 - ensuring that payments are made and the completion date is met;
 - fixing rates for delays or extra work;
 - authorising time extensions for unavoidable delays;
 - acting as an arbitrator.
- The consultant will probably be paid by the client.
- Such consultants may be individuals with professional qualifications, e.g. a Chartered Member of the BCS, or they may be larger organisations e.g. <http://www.integra-associates.com>

Other forms of software contract

Shrink-wrap licensing (or rip-seal license)

- With off-the-shelf software, there is no opportunity for a signed licence agreement.
- The response of the software industry to this problem is the **shrink-wrap** licence or **rip-seal** license.
 - The contract is displayed on the packaging, visible through a clear plastic film. The terms are deemed accepted if the package is opened.
- Shrink-wrap licence purports to be a direct contract between the software producer and the consumer, quite separate from the contract of sale between the dealer and customer.
 - Can such a contract actually exist in law?

Enforceability

- A contract requires three elements; offer, consideration and acceptance.
 - The display of the licence on the packaging constitutes an offer.
 - Consideration relates to the fact that the licensee is paying a fee to use the software.
 - Acceptance may be indicated by breaking open the package.
- Shrink-wrap licences are problematic because the opportunity to read the terms often comes after the contract is made, i.e. after the customer has paid and the software is physically handed to the customer.
- An important principle in contract law is that it is not possible to unilaterally introduce new terms into a contract after it has been made.

Beta Computers Ltd vs. Adobe Systems Ltd 1996

- Beta Computers supplied Adobe Systems with software produced by Informix Software Inc.
 - The software had a shrink-wrap licence – ‘Opening the Informix S.I. Software package indicates your acceptance of these terms and conditions’.
 - Adobe claimed it had the right to reject the software until opened.
 - Beta sued for the cost of the software.
- The case was brought in Scotland, and decided under Scottish law. The judge decided that:
 - the supply of the software is not separate from the supply of the media, so it is a single unique contract,
 - and was entered into on opening the package (i.e. when the terms of the licence were accepted), not when the sale is made.
 - Adobe could return the software
- But the decision was controversial. In the USA, the situation is clearer:
 - a purchaser can reject the software, having read the licence terms
 - i.e. after opening the package.

Web-wrap/click-wrap licences

- Copyrighted works (e.g. apps) are now acquired online.
 - In contrast to shrink-wrap licensing, the purchaser can be presented with the terms of license before purchasing.
 - The copyright work is only made available to purchase if the ‘agree’ button is clicked.
 - Clicking the ‘agree’ button incorporates the terms of the license into the contract to purchase.
- Web-wrap licenses cannot normally be used to assign intellectual property rights or exclusive licenses:
 - Unless a legally-recognised digital signature is used, because a written and signed document is required.

Contracts between author and publisher

- Many companies publish software that has been developed by self-employed freelance programmers.
- **Q. Why do freelance authors publish their software through other companies?**
- The freelancer owns the copyright in the software, so will grant a licence to the publisher permitting them to market the product on the basis of a royalty payment.
- Normally, the licence will be exclusive, giving a single publisher the sole right to deal with the software. The publisher then has the same rights under copyright law as if it owned the copyright itself.

Payment

- The software author will be paid a **royalty** by the publisher, which may be a fixed sum for every unit sold or a percentage of the price charged for the software.
- Alternatively, the author may be paid a lump sum for an exclusive licence.
- Care should be taken when quantifying the amount of royalty payments in the contract:
 - If royalties are based on the price of the software, is this the retail price or the payment the publisher receives from a dealer?
 - Do dealers receive discounts for bulk purchases that could affect the amount of royalty payment?
 - What happens if the publisher does not try hard to market the software?
 - Can the author check the publisher's accounts at regular intervals?
 - How frequently will royalty payments be made?
 - Can the author terminate the contract if the publisher ceases to market the software?

Summary

- Contracts for bespoke software are complex and must be drafted with care – standard contracts can help.
- Shrink wrap licences purport to be a separate agreement between the software company and the purchaser, but legally they appear to be inextricably linked with the contract of sale.
- Software is sometimes written independently and marketed via a publishing house; the publisher usually pays a royalty fee to the author.