

Office of Government Commerce

Open Source Software

Guidance on implementing UK Government Policy



Office of Government Commerce

1. Introduction

In answer to a question from Geraint Davies MP on 22 July 2002, Cabinet Office Minister of State, Douglas Alexander MP, launched a new, more embracing, policy on the use of Open Source Software (OSS) in Government.

The decisions of this policy are as follows:

- UK Government will consider OSS solutions alongside proprietary ones in IT procurements. Contracts will be awarded on a value for money basis.
- UK Government will only use products for interoperability that support open standards and specifications in all future IT developments.
- UK Government will seek to avoid lock-in to proprietary IT products and services.
- UK Government will consider obtaining full rights to bespoke software code or customisations of COTS (Commercial off the Shelf) software it procures wherever this achieves best value for money.
- UK Government will explore further the possibilities of using OSS as the default exploitation route for Government funded R&D software.

The policy publication goes on to say that the following **actions** will be taken to implement this policy.

- OGC will update their procurement guidelines to reflect this policy.
- Advice will be made available to all those involved in procurement exercises on areas of the software infrastructure and application marketplace where OSS has strengths and weaknesses.
- Advice will also be made available to all those involved in procurement exercises on how to assess the merits of OSS v proprietary solutions in procurements.
- OeE and DTI will discuss with academic research institutions the possibilities of future R&D work.

The key driver to this policy is that **Government is explicitly declaring its support for a level playing field between OSS and proprietary software procurement within Government by acknowledging the competitive viability of OSS solutions.** The policy also stresses the need to avoid lock-in to proprietary IT products and services, promoting more re-use in public sector ICT through obtaining, where appropriate, full rights to bespoke and customised software code and developing the position of OSS within Government funded R&D software.

It is not Government's intention to prefer, or be seen to prefer, one form of software over another. **The key test in all procurements must be that the solution procured offers value for money for Government.** As existing guidance points out,

value for money is defined as "the optimum combination of whole-life cost and quality (or fitness for purpose) to meet the user's requirement." This is not synonymous with lowest price. Procurement decisions will continue to be made on a case by case basis.

This Guidance note offers advice to Departments and Agencies on the practical implications of taking this policy forward. In particular it outlines some of the issues which need to be considered in all future software procurements. This note should be read in conjunction with existing guidance on UK Government Procurement Policy (see http://www.ogc.gov.uk/embedded_object.asp?docid=1332 for a list) and published OGC guidance notes, particularly the notes on "Value for Money Evaluation in Complex Procurements" and the OGC guidance on acquisition of IPR in procurements.

2. Open Source Software Definition

Open Source Software (OSS) is:

"Software where the source code (the language in which the program is written) is freely distributed with the right to modify the code, and on the condition that redistribution is not restricted, and indeed is obtainable for no more than the reasonable cost of reproduction".

In contrast, vendors of closed, proprietary, software provide only executable binary code, and not the human readable source from which that code is derived. Proprietary software vendors usually also place very specific limits on redistribution of the software.

The full, formal definition of OSS published by the Open Software Initiative is available at: <http://www.opensource.org/docs/definition.html>

The Free Software Foundation (www.fsf.org) also publishes a variation of the definition where it refers to "free software" rather than Open Source Software. The FSF definition stresses that "free software is a matter of liberty, not price." In this context, "free" means "freedom" as in free speech, not free as in "free beer".

Neither the OSI nor the Free Software definition implies anything about the cost of the software. Both definitions and the licences under which the software is distributed permit distribution to be charged for. In fact the major distributions of the best known OSS operating system known as GNU/Linux are sold commercially by distributors.

Note that most OSS is distributed under one of the main "Public Licences" such as the GNU General Public Licence, the Apache Software Licence or the Berkeley Licence. The essence of these licences is that the software is copyright but that the recipient has the right to read, modify and redistribute the code in either source or compiled form so long as the same licence and its rights and obligations are passed on to subsequent recipients. Thus, if a commercial organisation modifies OSS protected by one of the Public Licences before redistribution, then that redistributed software must carry the original copyright notice together with a notice that the software has been modified. The source code of both the original software and the modified version must be made available to the purchaser for no more than reasonable distribution costs.

This approach is directly at variance with the normal commercial practice of software distribution only in compiled form where the purchaser normally pays a charge for use of that software. The purchaser of closed source proprietary software almost never

acquires any right to see or modify the source code. In some circumstances, vendors of closed proprietary software will permit “shared” access to source code with customers. Such access usually comes under strict agreements designed to protect the vendor’s intellectual property. This is not an example of OSS as defined here.

3. OSS Benefits, Strengths and Weaknesses

3.1 Introduction

The best open source is demonstrably at least as good as, or better than, proprietary software by the only real test - it is in widespread operational use in the real world, particularly in infrastructure deployments such as operating systems and web servers. But the current usage of OSS outside these areas is patchy at best and is often limited to niche or specialist areas such as developer tools (compilers, scripting languages etc). There are some OSS desktop products which could offer serious alternatives to existing proprietary office suites, particularly where the full functionality or tight integration offered by proprietary products is not required, but OSS is not currently strong in this area.

3.2 Benefits

The main benefits offered by OSS are:

- It tends to have strong support for open standards for interoperability. This complements UK Government's insistence on interoperability standards through mandatory compliance with e-GIF.
- OSS is supplier independent. This is attractive in cases where Government would normally insist on a software escrow agreement. If OSS is used then the source code is already available to the purchaser (as well as the wider community) and Government could simply take that code to a new supplier should the original supplier disappear or withdraw support for whatever reason.
- OSS has lower licensing costs. Most OSS is distributed under one of a range of "Public Licences" which mandate that the code should be available to all for little more than the cost of distribution. However, purchasers should note that OSS is not synonymous with "free software". Some free software is actually distributed in binary (compiled) form in the same way as is proprietary software. One of the main benefits in greater use of OSS may arise from the pressure that OSS's lower licensing costs puts on proprietary software licenses.
- OSS tends to be portable to a wide range of platforms. Even where a particular platform is not currently supported, the open availability of the source code allows the community to port that software to a new platform relatively quickly. Platform independence gives the purchaser a wider choice of hardware in any procurement.
- Patches or updates to OSS following discovery of faults such as security vulnerabilities tend to be produced very rapidly, often within hours or days of announcement of the discovery of a problem. Users of closed source proprietary software are dependent upon the software vendor to release an upgraded binary version. This process can take considerably longer, leaving the user vulnerable to known threats.
- Avoidance of proprietary lock-in. OSS tends to be written to be independent of any other related product. Purchasers can often assume that one software product will work best with another from the same family produced by the same manufacturer. This can lead to what Qinetiq characterised as the "monogamy vortex". Use of OSS

offers users greater freedom to purchase other products, thus reducing this tendency for lock-in.

3.3 OSS Weaknesses or Drawbacks

Departments should be aware of some of the inhibiting factors affecting decisions to adopt OSS in any procurement. Such factors, which this Guidance is in part intended to address, may include:

- Uncertainty as to what exactly constitutes OSS and what its relative strengths and weaknesses are.
- Fear that support can be fragmented or difficult to obtain, particularly for niche products. That fear may now be misplaced. Departments should note in particular that many large suppliers such as IBM, Sun, and HP are investing considerable effort in supporting the GNU/Linux operating system. Further, many proprietary software suppliers are porting their applications to run under GNU/Linux. We expect this trend to continue and accelerate.
- Misunderstanding of the licensing and IPR implications of using or purchasing OSS. This is addressed below.
- Difficulties in identifying appropriate OSS applications for particular business problems. Because OSS is not "advertised" in quite the same way as is proprietary software, Departments may not be aware that a particular OSS product is available to meet their needs or that a product exists which could meet their needs after minor tailoring. In part this can be addressed by Departments themselves sharing experiences through membership of the new OSS special interest group (SiG) chaired by OGC. But the problem can also be mitigated by purchasers specifically pointing to the new policy in all new ITTs and stating that OSS solutions will be welcomed alongside proprietary ones. Intended contracts with Systems Integrators should be a particularly fruitful avenue for new or innovative uses of OSS in future. Integrators can be "technology neutral" in their approach and many may welcome the chance to reduce their own dependence on proprietary product sets. Departments should, however, be aware that many Systems Integrators are themselves effectively "locked in" to certain proprietary product sets by their need to maintain an in-house skill base capable of supporting that product. Maintaining a diverse skill base can be expensive and there is a natural tendency for Specialist Integration houses (such as major database design specialists) to focus their attention on a particular product or product set.
- Documentation can be idiosyncratic or sometimes non-existent. OSS developers are primarily motivated by exciting and/or innovative software development. They are less motivated to produce the sort of documentation that commercial buyers expect to see accompanying software. Where Departments purchase systems which have a significant element of OSS embedded in them it is reasonable to request business-like documentation as part of the contract. It should not be sufficient for the supplier to simply pass on the existing, sometimes sparse, documentation. Here the supplier or Systems Integrator should be encouraged to add value to the OSS by passing that documentation back out to the community. Most of the major OSS projects maintain an "Open Source Documentation" Project with the aim of fully documenting the software for ease of use. See for example the Linux Documentation Project at www.tldp.org.
- Lack of real world experience and support for migration from closed proprietary software installations to OSS. Procurements are rarely made in "green-field" sites. Purchasers must be able to integrate OSS with their existing installed base and must

understand how to migrate from a single supplier product set to a more diverse product set. This can be difficult enough when migrating from one proprietary product set to another where the purchaser has the wholehearted, and self-interested, support of the new supplier. It may be even more problematic when the purchaser is contemplating moving to an OSS product where there may be no single support point.

- OSS software often lags behind proprietary software in support for new hardware. Largely this is the result of the hardware suppliers themselves not releasing new drivers to the OSS community on time. But, as with issue of maintenance and support, that problem is diminishing as the hardware suppliers themselves begin to use and market OSS on their platforms.

4. OSS usage

OSS is not ubiquitous. Aside from its strong presence in developer tools such as compilers, interpreters and scripting languages, areas where it is particularly strong are currently mainly infrastructure related, i.e. operating systems, firewalls, application level relays and proxy servers, web servers, file and print servers, domain name servers, mail transport servers and network news transports.

However, the following figures drawn from publicly available surveys in mid 2002 give an indication of **the dominance of OSS in public facing internet services:**

- Apache is the number one **web server** at nearly 60% of all installations. Second place was held by Microsoft at just under 30%. (Source Netcraft)
- GNU/Linux is the number two **operating system** at nearly 30% of all servers behind Microsoft operating systems at nearly 50% (Source Netcraft)
- Sendmail (a mail transport agent) was the number one **mail server** at 42% of servers polled. Microsoft Exchange was second at 18% (Source DJ Bernstein)
- "BIND" the Berkley Internet Naming Daemon responsible for mapping domain names to IP addresses ran on 95% of all public reverse **DNS servers**. (Source Information Sciences Institute of the University of Southern California).

As yet, with the exception of some niche areas such as graphics manipulation, OSS is not strong in the application arena (such as Database Management Software, ERP or Financial Systems). Nor do we believe is it yet likely to be a viable alternative to the dominant product sets on the desktop where tight integration between products is seen as beneficial. However, as noted above, we are seeing a strong trend towards suppliers of application software porting their applications to Open Source Operating systems such as GNU/Linux. **It is thus increasingly likely that Departments may find themselves purchasing systems which include an OSS element.** We are also beginning to see strong OSS alternatives to the dominant desktop and office software and we expect this trend to continue and, indeed, accelerate.

5. IPR issues

The new OSS policy specifically states:

"UK Government will obtain full rights to bespoke software code that it procures and all customisations of COTS (Commercial Off-the-Shelf Software) packages that it uses wherever this achieves value for money."

Existing guidance issued in January 2002 by OGC says, inter alia, that:

"Decisions should be reached by balancing the costs and benefits of acquiring IPR against those of leaving it with the private sector contractor and, where appropriate, acquiring the right to use IPR ..."

"In short ... within the over-arching policy of achieving VFM, the guiding principle should be that in all cases ... IPR resides with the party who is best able to exploit it."

OGC legal advice is that it could be argued that these two "tests" may not always produce consistent answers. That is, it may be the case that the private sector is best able to exploit the IPRs whereas the public sector obtains best VFM by owning the IPRs (perhaps for the very reason that the private sector may wish to exploit the IPRs in a way that may be detrimental to the public sector).

Before deciding to seek to acquire full rights to any software, purchasers are advised to decide whether there are any rights to acquire, what those rights might be and whether there is any benefit to the public purse by seeking to acquire such rights.

This decision may be guided by the following three steps:

Step 1 - The Technical Assessment

The technical analysis involves:

- **identifying what is actually being written or created or developed in the course of the project; and**
- **deciding whether it is new, novel and original.**

These questions must be answered in order to determine whether, in the course of the project, or procurement it is likely that IPRs may arise and if so to identify the relevant deliverables/assets. As a general rule, IPRs only subsist in, and protect, new, novel or original material.

Step 2 - The Legal Assessment

Whilst it may be convenient to refer collectively to certain intangible property as "intellectual property", this is simply a collective term for a large body of law that differs in many respects when one considers the detail. Therefore IPRs need to be considered in the context of:

- what has been written, created or developed (see Step 1 above); and
- having regard to that, which category (or categories) of intellectual property law is or

are relevant. The most likely are copyright, design rights, database rights, patents and/or confidential information or know-how.

Having made this assessment, the legal analysis then needs to determine:

- **whether any IPRs subsist in the deliverables or assets that will be written, created, or developed; and**
- **which party owns those IPRs.**

The different categories of Intellectual Property law may require different tests to be applied in answering either or both of these questions. Note that IPR ownership is not normally an economic right, IPR is normally vested in the person or body who creates the work, not the person or body who pays for that creation unless the contract specifically states otherwise.

The legal analysis will, therefore, determine whether (and which) IPRs are relevant for the purposes of Step 3.

Step 3 - The Financial/Economic Assessment

This involves an analysis of the likely balance between benefit(s) to be derived from the cost of ownership versus the benefit(s) of leaving IPR ownership in the private sector. Such an assessment will almost certainly need to be made on a project-by-project basis. But the range of issues to consider (some of which may be difficult to quantify in strict money or "value" terms) will include:

- The competitive advantage sought by the public sector in owning the IPRs;
- The commercial value of the IPRs in licensing terms: (Are there any potential licensees of the IPR and what would they be willing to pay for a licence?)
- The cost of ownership:
 - Registration costs - where applicable;
 - Enforcement costs - taking action against third party infringements to protect the IPRs;
 - Defence costs - defending one's property if the validity of the IPRs is challenged, or if it is alleged that there is an infringement to a third party's IPRs; and
 - Administration costs - policing the commercialisation of the IPRs (e.g. compliance by licensees with licence terms).
- The possibility of deriving income (value) through contractual provisions other than acquiring ownership of the IPRs, e.g. agreeing a third party revenue distribution arrangement with the private sector supplier and finally,
- The premium, if any, charged by the private sector for transferring IPRs to the public sector; or, conversely, the discount (if any) offered by the private sector for not transferring the IPRs to the public sector.

Purchasers should attempt to decide whether there may be value in acquiring IPR in any software developed in the course of procurement at as early a stage in the procurement process as possible. If they do decide that there may be some merit in acquiring such rights, they may wish to seek specialist legal or contractual advice at that stage.

6. Avoiding Proprietary lock-in

Government needs to be particularly wary of making investments which are specific to a particular supplier or partner which could lock it in for the economic lifetime of the investment. This applies not just to software procurements but equally to partnerships with outside suppliers of goods and services.

Departments and Agencies must first identify where there is potential for lock-in and should differentiate between those areas where the potential is for current technological lock-in (such as can happen with commitment to a particular storage medium for example) but there are multiple vendors for that technology, and those where the potential is simply for commercial lock-in to a particular provider (such as can arise from the use of a single vendor product where data is stored in a proprietary, closed format).

Proprietary lock-in can arise for a variety of reasons:

- The perception that one product from a supplier will necessarily provide better integration with existing products than will a competitor. This process can be particularly insidious when seen over time. Purchasers can often unconsciously limit the choice of software to certain product sets because of perceived issues of compatibility with existing installed software.
- Inappropriate contractual commitments such as an agreement to source from a particular supplier in return for a discount. The value of that discount may not be sufficient to offset the cost of later lock-in.
- Brand specific training (which can impact on transition costs). Note that this can occur both within the purchasing organisation which may have invested heavily in a particular product set currently in use and in any outsourcing partner which may have standardised on a particular product set in order to minimise its own costs.
- Tendencies to assume without detailed analysis that switching costs may be too high to make transition viable at contract renewal.
- "Laziness". There are costs in identifying alternative product sets. This may be particularly relevant in the case of attempts to change from a dominant proprietary product to an OSS alternative because market awareness of OSS products is not as mature as that for proprietary products supported by advertising;
- Supplier "loyalty programs" such as reduced maintenance or upgrade charges in return for future commitments to purchase.
- Acceptance that current market dominance necessarily implies "best option".

Purchasing Units should question all assumptions about maintenance of the status quo or acceptance of brand leadership or dominance before purchase. In particular, purchasers should measure the degree of lock-in in terms of the cost of escape from that lock-in. A balance needs to be struck between the costs of escape or transition at the end of one contract and any perceived benefit in continuing to accept a closed, proprietary product.

To some extent, mandatory adherence to eGif for interoperability should mitigate the effects of proprietary lock-in. As the Qinetiq report on OSS noted, it is the use of proprietary standards and protocols that effectively mandates the purchase of further products from the same supplier, leading to what Qinetiq call the "monogamy vortex". **Purchasers must satisfy themselves, and Gateway reviewers, that full cognisance has been taken of the potential for such lock-in and that any decision to continue with a proprietary standard can show that it achieves best value for money over the full life cycle.**

In addressing the issues of lock-in, purchasers may find it helpful to ask the following questions:

- **Is the tight integration between one product and another offered by a supplier actually a requirement of the product or service being purchased?**

If not, can it be weighted as a desirable option rather than mandatory? Will a lower level of integration offered by a competitor product meet the requirements? If tight integration is necessary, is this a reflection of past decisions which have led to current effective lock-in? Is it desirable to continue that lock-in, making later exit costs higher or is there now a demonstrable case for buying out of the lock-in for a future benefit (invest to save)? Have the full exit costs been identified? Have the likely future exit costs after this latest procurement been identified?

- **Is the added functionality offered by a new product set from an existing supplier actually a requirement?**

If so, is that added functionality required across the whole organisation? If not, is there a core set of users who need the new functionality whilst the majority may be content with existing functionality? (For example, system managers or administrators or specialist users such as finance officers may need particular new functions, but the majority of the organisation's staff may not). Beware of solutions which "extend" or "enhance" particular standards in order to provide added value or better integration with existing product sets from a single supplier. Ask whether that "added value" is necessary or beneficial both in the short term and the longer term when the Organisation may wish to move to an alternative product set.

- **Is the value of any contractual "special offer" worth the possibly higher future cost of exit from the existing proprietary offering?**

Has the full cost of exit been quantified?

- **Have the real costs of re-training been identified?**

Purchasers should particularly be wary of assumptions that re-training costs on the move from a proprietary solution to one based on an OSS solution outweigh any benefits to be derived from the transition. Similar costs will arise in any move from one closed proprietary product to another. Even maintaining the status quo will attract re-training costs as single suppliers upgrade their current offerings.

- **Have the other costs of transition (hardware refresh, purchase of new tools etc) really been identified and quantified?**

Beware of untested assumptions about transition costs. For example, there may be an assumption that moving to an OSS solution will imply new hardware when in fact staying with a proprietary solution will force a hardware refresh that would **not** be required when moving to an OSS solution. Further, because OSS often has a lower hardware requirements than the equivalent proprietary software, significant savings can be made in hardware costs over the lifetime of the new procurement. Ask prospective suppliers to specify the optimum hardware requirement for their preferred solution.

- **Is the value of any licence or maintenance charge discount given for a "loyalty program" or extended contract life sufficient to make it worth staying with a proprietary solution?**

Bear in mind that most analyses show that licence costs amount to no more than 10% of the Total Cost of Ownership of a software product. A 10% reduction in that licence charge may appear superficially attractive, but look less so under critical analysis.

- Current market dominance in a particular sector should not necessarily be taken to mean that that product or product set offers the best value for money. Market position changes over time. Accepting the current dominant product on the grounds that it is the de-facto standard can leave purchasers exposed to what Qinetiq characterised as "data dungeons" - i.e. the data they own is locked in to a proprietary storage format. Purchasers should satisfy themselves that either the dominant products are capable of storing and/or exporting the data in an open standard format (such as XML), or that there are sufficient third party tools and products capable of data conversion to intermediary formats. Again be wary of products which offer proprietary extensions to published open standards since such usage can mean that later data export is impaired and/or could be particularly costly.

This guidance note will be revised and updated in the light of experience gained in new procurements following publication of the new policy, Purchasing Units are particularly encouraged to provide feedback on the value or otherwise of this guidance as part of their Gateway review process.

OGC
September 2002

Open Source Software

Open Source Software

Office of Government Commerce
Rosebery Court
St. Andrews Business Park
Norwich
NR7 0HS

Telephone: 0845 000 4999
Fax: 01603 704817
e-mail: ServiceDesk@ogc.gov.uk
www.ogc.gov.uk



Office of Government Commerce