# NZGOAL Software Extension

New Zealand Government

# Contents

# Introduction

## Purpose

1    Government agencies invest significantly in software development and often own the copyright in the software that they develop or that is developed for them. Public sharing and the licensing of this government-owned software under free and open source software licence terms has the potential to:

(a)    save agencies time and money, resulting in a more efficient use of scarce resources;

(b)    encourage open innovation on the part of both the public and private sectors;

(c)    contribute to economic growth, primarily through the private sector being able to leverage and support government investment in the software it openly releases for re-use;

(d)    contribute to the formation of trusted communities of users whose public and private sector members have common or similar goals or interests;

(e)    result in continuous and ongoing maintenance of the released software code through these communities of users in a way that may not be achievable by a single agency alone;

(f)    enable agencies to better align their operational and strategic activities with relevant aspects of the Government ICT Strategy 2015; and

(g)    in some cases, foster transparency – for those who can read software code – as to the methods or algorithms used for the creation or delivery of public data and services, thereby enabling critical analysis and potentially the provision of improvements back to the releasing agency.

2    This NZGOAL Software Extension (**NZGOAL-SE**) provides agencies with a means of realising this potential. It:

(a)    explains the legal and policy context that is relevant to agencies' open source licensing of software;

(b)    sets out a series of policy principles to guide agencies in their open sharing of software code;

(c)    advocates the use of particular open source software licences for this purpose; and

(d)    sets out a review and release process to guide agencies through the review of the software they propose to release for re-use, the purpose of which is to help agencies make decisions that are legally robust and practically useful.

## Free and open source software and licences

3    For the purposes of NZGOAL-SE, free and open source software (**FOSS**) is software source code that is released on licence terms that grant others the freedom to use, copy, modify and distribute the software, for either non-commercial or commercial purposes, as long as they comply with the applicable licensing conditions. It is important to note the following:

(a)    the licensing conditions that users of the software need to comply with depend on the form of FOSS licence applied to the software;

(b)    "free" in FOSS refers to the granting of a set of "freedoms" or permissions usually only available to the copyright owner and arising from the bundle of property rights the owner has in its copyright work; and

(c)    there are several common phrases in use that refer to the same style of licensing, such as "free software", "free and open source software", "open source", "FOSS", "OSS" and "FLOSS". Depending on the context, NZGOAL-SE uses a combination of "free and open source software", "FOSS" and "open source", all with the same meaning referred to above.

4    There are two main forms of FOSS licence:

(a)    **more permissive licences** that confer broad freedoms and minimal obligations on those who wish to use, adapt and distribute the software (e.g., the MIT licence, the BSD licence and the Apache 2.0 licence); and

(b) **sharealike licences** (also known as copyleft licences) that confer similar freedoms and require those who adapt the licensed software to license their adaptations with the same licence if they distribute them (e.g., the GNU General Public License, or GPL for short).

## Approach and scope

5    NZGOAL-SE is an extension of, and is modelled in part on the approach to, the open licensing of government copyright works set out in the New Zealand Government Open Access and Licensing framework (**NZGOAL**). NZGOAL-SE is a self-contained extension or framework, rather than being part of the original NZGOAL framework, because certain considerations are different to those relating to open information and data. Incorporating both frameworks under a single roof would unduly complicate matters for agencies and others interested in the frameworks.

6    NZGOAL-SE is not concerned with the proprietary versus open source debate or with the considerations that agencies may wish to take into account when using existing open source for their own internal purposes.[1] Its sole focus is on the public release and open source licensing by agencies of software they own or are authorised to release and license.

## Additional guidance notes

7    Additional guidance notes may be released over time which:

(a)    explore, in greater detail, some of the issues addressed or raised in NZGOAL-SE; or

(b)    address operational or technical issues which arise in practice.

---

1    On these topics, see the Australian Government's *A Guide to Open Source Software for Australian Government Agencies* (version 2.0, June 2011) at http://www.finance.gov.au/files/2012/04/AGuidetoOpenSourceSoftware.pdf

# Legal and policy context

## Copyright law

8    Key aspects of copyright law relevant to open licensing generally are set out in the NZGOAL Copyright Guide.[2] Additional aspects relating specifically to software include the following:

  (a)   **copyright exists in original software**: software is a form of "literary work" and original literary works are a category of works in which copyright exists (section 14(1) of the Copyright Act 1994 and the section 2(1) definition of "literary work");

  (b)   **infringement**: as such, unless entitled to do so by a copyright licence or statutory provision, a person infringes copyright in software that that person does not own when he or she does any of a number of "restricted acts", the most common of which is copying the work or a substantial part of it or adapting the work;

  (c)   **adaptations**: an "adaptation", in relation to a literary work that is a computer program (i.e., software), includes "a version of the program in which it is converted into or out of a computer language or code or into a different computer language or code, otherwise than incidentally in the course of running the program" (section 2(1));

  (d)   **first owner of copyright**:

      (i)    **employee-employer relationship**: when employees create software for their employer in the course of employment, the employer is the first owner of copyright in the software unless the employment contract states otherwise. This is the case where the employer is the Crown (section 26) and where the employer is another person, agency or entity (section 21(2));

      (ii)   **service provider-customer relationship**: when a service provider creates software for a customer under a contract for services, the customer is the first owner of copyright in the software unless the contract for services states otherwise. Again, this is the case where the customer is the Crown (section 26) and where the customer is another person, agency or entity (section 21(3)).

## Guidelines for Treatment of Intellectual Property Rights in ICT Contracts

9    These Guidelines[3] are relevant where an agency procures the development of software from a service provider. They state that only in limited circumstances should the government own and exploit new intellectual property rights (**IPR**) created under an ICT contract. The default position is that the supplier should own the new IPR, with licences being granted to the customer agency and other State Services agencies.

10   This position could create an obstacle to an agency licensing developed software on open source terms because the agency can only do so if it has the requisite rights to do so. If it doesn't own the copyright, it doesn't have the requisite rights unless expressly authorised by the copyright owner.

11   The Guidelines recognise, however, that there may be situations where an agency needs to own the IPR. One of those situations is where the agency "intends to allow free use of the IPR on open source terms". Agencies would, therefore, be acting consistently with these Guidelines if they were to require the development contract to confer ownership of copyright in the software on the agency, on the basis that they wish to license the software on open source terms.[4]

---

2    See https://www.ict.govt.nz/guidance-and-resources/open-government/new-zealand-government-open-access-and-licensing-nzgoal-framework/ss-and-licensing-nzgoal-framework-nzgoal-copyright-guide/

3    The Guidelines are available at https://www.ict.govt.nz/assets/Uploads/Documents/ipr-guidelines-2008.pdf

4    Where the agency owns the copyright, it could still grant a non-exclusive custom licence to the service provider, before or at the same time as releasing the software on open source terms, that enables the service provider to use the software for other purposes. For example, the service provider may wish to be able to use the software it has developed for other projects before the agency releases it on open source terms.

**Government Rules of Sourcing**

12      Who is to own the copyright in new software is a significant issue that needs to be considered at the time of procuring the software development services, a point that is made clear in Rule 61(1) (Intellectual Property) of the Government Rules of Sourcing.[5] That Rule states:

> "If an agency's procurement of goods, services or works involves the supplier creating new Intellectual Property, the agency **should** set out, in its Notice of Procurement, its intentions regarding ownership, licensing, and future commercialisation of that Intellectual Property."

---

5      The Government Rules of Sourcing are available at http://www.business.govt.nz/procurement/for-agencies/key-guidance-for-agencies/the-new-government-rules-of-sourcing

# NZGOAL-SE Policy Principles

## Introduction

13    Government agencies are strongly encouraged to apply the following Policy Principles in relation to publicly releasing and licensing their software on open source terms. The Principles should be read in their entirety but here is a summary:

| Policy Principle | Summary |
| --- | --- |
| Open access and public release of agency software using free and open source licences | If an agency wishes to license its software, it is encouraged to do so on open source terms, to use public version control and source code repository platforms and to follow the NZGOAL Review and Release Process and decision tree. |
| Ensuring copyright ownership or right to sub-license | Make sure your agency has the required copyright-related rights to license the software. |
| Exceptions | Releasing under open source terms does not apply where an exception applies. |
| Adaptations | Respect existing licenses when re-using existing software. Be careful when considering whether you're "adapting" pre-existing software. |
| Security code review | Consider whether a security code review is required to identify sensitive elements that may need to be removed. |
| No additional controls or discrimination | Do not seek to impose requirements that are inconsistent with the freedoms in the chosen FOSS licence. |
| No charging | Do not charge people for access to FOSS licensed software. |
| Updating FOSS licensed software | If you spot a bug with software you've licensed, consider informing users and, if you fix the bug, release the updated files. Take appropriate action if there's a security risk. |
| Contributions | Where appropriate, adapted, pre-existing open source licensed software should be contributed back to the open source community. Agencies should provide guidance on how they will accept external contributions to their own released source code. |
| Obtaining rights when procuring or commissioning the development of software | If you commission the development of software that you want to FOSS license, make sure you obtain the rights to license it on FOSS terms. |
| Act fairly towards developers when drafting IP warranties and indemnities | If you commission the development of software that uses pre-existing third party FOSS licensed software and/or that you may FOSS license, act fairly in relation to IP warranties and indemnities. |
| Liability | Replicate all disclaimers and exclusions contained in the relevant FOSS licence when you release software under that licence. |
| Review and Release Process | Follow the NZGOAL-SE Review and Release Process. |

Each Policy Principle is set out below.

**Open access and public release of agency software using free and open source licences**

14    If government agencies have the required copyright-related rights to do so, and wish to license their software source code for re-use, they are encouraged:

   (a)    to consider making their source code, that is or may be of interest or use to people, available for re-use under a FOSS licence as further described in these Policy Principles and in accordance with the NZGOAL-SE Review and Release Process and decision tree, unless an exception in paragraph 18 applies; and

   (b)    if they decide to do so, to:

      (i)    use existing version control systems and source code repository platforms to allow for discussion and improvement of released software;[6] and

      (ii)    make the software publicly accessible by releasing it online; and

      (iii)    use existing, freely available software to interact with the released software source code

   (the **Open Access and Licensing Principle**).

15    For the purposes of NZGOAL-SE, the recommended set of free and open source software licences for which selection guidance is provided are:

   (a)    the **MIT licence:**[7] this is a more permissive licence in the sense that it does not contain a sharealike obligation that would require adaptations to be licensed under the same terms if distributed (the full text of the MIT licence can be found on the website of the [Open Source Initiative](#))[8]; and

   (b)    the **GPL licence:** this is sharealike/copyleft licence that requires those who adapt the source code to release their adaptations on the same terms if they distribute them (the full text of the GPL licence can be found on the website of the [Free Software Foundation](#)).[9]

**Ensuring copyright ownership or right to sub-license**

16    Paragraph 14 makes it clear that, for the Open Access and Licensing Principle to apply, an agency must have the required copyright-related rights to license the software.

17    Agencies should only license software for re-use by others under a FOSS licence where:

   (a)    they own or can obtain an assignment of all copyright in the software and have not exclusively licensed it to a third party; or

   (b)    to the extent they do not own the copyright, they have or can obtain permission from the copyright owner(s) to do so

   (the **Rights Clearance Principle**). It can be important, in this context, to check whether developers have reused tracts of code from elsewhere and, if they have, to assess the licences that apply to those tracts to ensure the agency has all the rights it needs.

---

6    Both the United Kingdom Government Digital Services ([https://www.gov.uk/service-manual/making-software/choosing-technology.html#sharing-software](https://www.gov.uk/service-manual/making-software/choosing-technology.html#sharing-software)) and the United States Federal Government ([https://sourcecode.cio.gov/Implementation](https://sourcecode.cio.gov/Implementation)) express the value of modern software development practices such as version control and code repository platforms for sharing and the improvement of publicly released government source code.

7    The MIT licence is considered preferable to the BSD 3-Clause licence (another commonly-used permissive open source software licence) because the MIT licence includes a clearer grant of rights and expressly includes the right to sub-license (the BSD licence does not). Whilst many interpret a right to sub-license as being implicit in the BSD licence, the absence of an express reference to it in the BSD licence could produce uncertainty for users, most notably users who wish to incorporate government-produced code in a work licensed under the GPL. The Free Software Foundation considers the BSD licence to be compatible with the GPL but that must depend on a particular interpretation of the BSD licence wording (see generally A Sinclair "License Profile: BSD" IFOSS Law Review, 2(1) pp. 1-6). The MIT licence doesn't contain the BSD's 'no endorsement' clause but, in most cases, the law would prevent claims of endorsement without permission anyway. Another common permissive licence, the Apache License 2.0, was not chosen because it is more complex. The Free Software Foundation considers it preferable to the BSD and MIT licences as it deals with patent licensing and prevents 'patent treachery'. However, in New Zealand the Patents Act 2013 excludes computer programs "as such" from patentable subject matter (and, in any event, historically government agencies have not generally been in the business of applying for software patents). NZGOAL-SE could have recommended its own bespoke permissive licence instead of the MIT licence but that would have contributed to further licence proliferation and exposed developers to a licence they're not familiar with.

8    See [https://opensource.org/licenses/MIT](https://opensource.org/licenses/MIT)

9    GPL as used in NZGOAL-SE refers to the GNU General Public Licence version 3 or later. See [http://www.gnu.org/licenses/gpl-3.0.en.html](http://www.gnu.org/licenses/gpl-3.0.en.html)

**Exceptions**

18    The Open Access and Licensing Principle does not apply where:

   (a)    **civil wrongs**: open source licensing of the software would constitute a breach of contract, breach of confidence, breach of privacy, disclosure of a trade secret or other actionable wrong;

   (b)    **commercial interests**: open source licensing of the software would be contrary to an agency's own or the Government's legitimate commercial interests (note, however, that most government agencies are not in the business of commercialising developed software);

   (c)    **security or privacy risk**: release of the software on open source terms would create an unacceptable security risk (whether to an agency, organisation or individuals) or an unacceptable privacy-related risk.

**Adaptations**

19    In the course of its activities, an agency may incorporate or adapt pre-existing software source code that is licensed under one or more FOSS licences. Alternatively, an agency may produce new software source code that interacts with a pre-existing FOSS-licensed application. Take a plug-in or extension to an existing application as an example. Depending on the context, a plug-in or extension may be an adaptation of pre-existing source code (e.g., where it has forked an old plug-in or extension) or it may be new source code designed to interact with a pre-existing application but without taking any pre-existing source code. If an agency proposes to release its source code in these kinds of situations, the agency should:

   (a)    respect the terms and conditions of the existing licences (such as attribution requirements); and

   (b)    where possible, use the same free and open source licence or licences applying to the pre-existing software source code, even in cases where - strictly speaking - it is not required to do so

   (the **Respect Existing Licences Principle**).

20    It is important to note that using more permissively-licensed source code (e.g., MIT-licensed source code) and sharealike/copyleft-licensed source code (e.g., GPL-licensed source code) together in a broader project may require you to release your new source code under the same sharealike/copyleft licence or a compatible licence (in most cases the licence will be version 2 or 3 of the GPL)[10] rather than under the more permissive licence. Whether this is the case depends on the circumstances. Legal advice should be sought if required.

21    Difficult legal questions can arise as to whether a given piece of developed software, that in some way leverages or interacts with other software, is in fact an adaptation or derivative of that software. If:

   (a)    an agency's developed software does leverage or interact with other software; and

   (b)    that other software, or part of it, is software owned by a third party that has been released under a sharealike open source software licence like the GPL,

   the agency should be cautious about concluding that its developed software is *not* an adaptation or derivative of the other software. The need for caution arises because if the agency concludes mistakenly that the developed software is not an adaptation, release of the developed software under an incorrect licence could result in the agency breaching third party rights and expose it to the risk of complaint or legal action. End users would also be at risk of breaching third party rights and be exposed to the risk of complaint or legal action. If an agency is in any doubt on this issue, it should seek specialist advice (which could be both technical and legal) before releasing the software for re-use.

_____

10    Whilst versions 2 and 3 of the GPL differ in certain respects, in essence the GPL allows people to copy and distribute the software, to charge a fee for transferring it or providing warranty protection, and to modify the software and distribute resulting derivative works. And, if a person distributes a derivative work, that person needs to license it under the GPL, otherwise that person's licence to use the software will terminate. The full text of versions 2 and 3 of the GPL (both versions are in common use) can be found on the website of the Free Software Foundation at http://www.gnu.org/licenses/gpl-3.0.en.html.

**Security code review**

22    Before releasing software for re-use, agencies should:

    (a)    consider whether there is any prospect that the code or associated files may contain sensitive elements that may need to be removed prior to release; and

    (b)    if there is such prospect, have the code and any associated files security-reviewed before release.

**No additional controls or discrimination**

23    When releasing developed software under an open source software licence, agencies must not seek to impose controls or requirements, whether contractual or otherwise, that are inconsistent with the freedoms or permissions granted by the selected licence. For example, agencies are *not* able to license software under an open source software licence and then seek to discriminate between individual, not-for-profit and commercial uses of the software.

**No charging**

24    Government agencies that release software under open source software licences should not seek to charge people for access to the software.

**Updating FOSS licensed software**

25    If an agency:

    (a)    publicly releases software on open source terms; and

    (b)    subsequently identifies a bug or other issue with the software that could have a material adverse effect on users of the software,

the agency should:

    (c)    consider whether to inform users of the software of the bug or other issue (e.g., by adding a notice to the repository, site or service that contains the software files); and

    (d)    if the agency has rectified the bug or other issue for its own purposes, release the updated file(s) to the relevant repository, site or service.

26    If the agency is aware:

    (a)    of a bug or other issue that creates a risk to the security of sites or services using the software or to the confidentiality or privacy of information held by those sites or services; and

    (b)    that other government agencies are using the software,

the agency should inform the Government Chief Information Officer (and where relevant the Government Chief Privacy Officer and Office of the Privacy Commissioner) as soon as possible, take all reasonable steps to inform the other agencies of the risk and give them time to mitigate the risk before making any public announcement that could result in people with malicious intent or crackers[11] exploiting the bug or other issue. Agencies may also wish to consider including a Coordinated Disclosure Policy[12] in an accompanying "contributing" file to set expectations as to how the reporting of security risks ought to be handled.

**Contributions**

**Agency contribution to FOSS communities**

27    Where an agency has re-used and adapted pre-existing free and open source software source code or has developed new code to interact with free and open source software source code, the agency should:

---

11    See the Internet Security Glossary, Version 2 on Crackers https://tools.ietf.org/html/rfc4949#page-84
12    Coordinated disclosures ensure there is clear and explicit process for both external contributors and agencies for the reporting of security bugs and issues in publicly released software. Guidance for the New Zealand context can be found on the NZITF website at http://www.nzitf.org.nz/pdf/NZITF_Disclosure_Guidelines_2014.pdf.

(a)     consider asking the free and open source software community or the software source code owners whether any agency adapted feature or improvements or the agency's new source code would generally be re-usable by others (if this isn't already clear);

(b)     if so, contribute their adapted or new source code back to the free and open source software community project unless an exception listed in paragraph 18 applies; and

(c)     avoid using a diverging fork[13] copy of the software source code when it could reasonably use the upstream source code and contribute agency adaptations of it back to the community for general re-use

(the **Contribute to FOSS Communities Principle**).

## Agency acceptance of contributions from others

*Contribution guidance and files*

28     Part of the value of releasing free and open source software in accordance with the Open Access and Licensing Principle is that external contributors can re-use and contribute back improvements to agency-released software source code. To accept contributions from external contributors in a manner that enables contributions to be licensed properly and shared with others, agencies should consider providing explicit contribution guidance to users that:

(a)     explains any technical aspects of making the contributions;

(b)     sets expectations for users as to how the agency will review and accept contributions of source code;

(c)     sets out any code of conduct, moderation policy or terms of use that apply when external contributors engage publicly in discussions about the agency-released software source code;

(d)     indicates whether external contributors need to physically sign and return a contributor licence agreement[14] or other document before the agency will accept contributions from them, or specifies that the act of contributing source code means the contributing user is agreeing to license his or her contributions to the agency and others under a specified FOSS licence; and

(e)     ensures that users know that they must own the copyright in their contributions or be permitted to license them to the agency and others under the licence that the agency specifies (this is important as it seeks to mitigate the risk of the agency accepting and publishing external contributions that, for copyright reasons, should not be accepted and published).

29     In the open source software context, a common means of providing this contribution guidance to users is by including a 'contributing' file in documentation that accompanies the source code. It is important that this file (or other document if an alternative approach is used) is brought to the attention of users before or at the time they submit contributions. Some source code repositories bring the existence of a 'contributing' file to the attention of users at the time of submission but if an agency is not using such a repository it should take care to ensure the terms of contribution are brought to users' attention.

30     Whilst this is a decision for agencies, NZGOAL-SE suggests that it will usually be unnecessary, administratively burdensome and potentially counterproductive to require contributing users to physically sign and return a formal contributor licence agreement or other document. In most cases agencies will be able to secure the rights they need, with sufficient confidence, by including the 'contributing' file mentioned above in documentation accompanying the source code and ensuring that this file is brought to users' attention before or at the time of submitting their

---

13     A diverging code fork occurs when agencies make changes to the code of open source software without publishing the code back to the software's development community. The fork is the split between the agency's version of the software and the version published by the community. Any further changes made by either the agency or the community will increase the differences of the source code. This can make it difficult for the agency to upgrade to a new published version, as the agency would have to reapply all its changes. This risk may be mitigated by contributing modified source code back to the open source software community if appropriate.

14     A Contributor Licence Agreement, or CLA, is an agreement that sets out the terms under which a person or entity makes contributions to the project or software. A CLA will deal with licensing terms and can also agree with topics such as warranties and indemnities.

contributions. Agencies should seek advice from their legal teams if required.

*Comment on assignment versus licensing*

31    Agencies may wish to note that there are two ways in which an agency could obtain the rights it needs from contributors. An agency could require contributing users to assign (i.e., transfer) any copyright in their contributions to the agency or it could require contributors to license their contributions to the agency and others under a specified FOSS licence. The advantage of seeking assignments of copyright is that this enables the party maintaining the source code to own it all. That makes it easier to take action against a user who does not comply with the applicable FOSS licence terms. The significant disadvantage of seeking assignments of copyright is that it can inhibit contributions (as developers may wish to retain their copyright). In a government context, it could also be seen as 'over-reaching'. An assignment also needs to be "in writing signed by or on behalf of the assignor" (section 114 of the Copyright Act 1994). It will be evident from the approach suggested above that NZGOAL-SE does *not* recommend that agencies seek to obtain assignments of copyright in users' contributions. Rather, the suggested approach allows contributors to retain their own copyright and only requires them to license their contributions under a specified FOSS licence.

*Suggested licensing text for 'contributing' file*

32    NZGOAL-SE suggests that agencies include the following kind of licensing text in their 'contributing' file or other documentation that they bring to the attention of users before or at the time of source code submission (the content in square brackets needs to be completed):

```
Copyright and licensing

The source code for this [name of project or software] is licensed
under the [insert name of applicable free and open source software
licence]. By contributing source code to this [name of project or
software], you are agreeing to license your contributions under and on
the terms of the [insert name of same licence]. Please note that your
licence is irrevocable and royalty-free. You or your licensors retain
any copyright in your contributions while allowing others to re-use
the source code in any way they like as long as they meet the
requirements of the licence.
```

33    Agencies may also wish to ensure that the 'contributing' file, or other document that is brought to the attention of users, contains the matters referred to in paragraph 28(a)-(e) above.[15]

## Obtaining rights when procuring or commissioning the development of software

34    When procuring or commissioning the development of software, government agencies should consider whether they may wish to release the software to the public for re-use under an open source software licence. If the answer is yes, agencies should consider the steps that may be required as part of their procurement and contracting processes to ensure that either:

(a)    they have the relevant rights to release the software under an open source software licence; or

(b)    that the developer will release the software under a specified open source software licence.

35    Such steps may include:

(a)    ensuring the agency owns the intellectual property rights in the developed software; or

(b)    ensuring the agency obtains a broad licence from the service provider allowing the agency to sub-license the software under the MIT licence or GPL (or, if required, some other free

---

15    The 'contributing' file could also seek to obtain a warranty (promise) from contributors that they have all the rights they require to submit their contributions on the terms of the specified licence and an indemnity in favour of the agency should it turn out that the contributor does not have the required rights. The indemnity clause would seek to enable the agency to recover, from a contributor, loss the agency incurs as a result of the contributor's breach of the warranty. NZGOAL-SE suggests, however, that including an indemnity could be unnecessary and/or counter-productive. Not only could it inhibit contributions but it could pose difficulties for contributing government departments given restrictions on the granting of indemnities under the Public Finance Act 1989.

and open source software licence); or

 (c) insisting on contractual provisions that require the service provider to release the software under a specified open source software licence (and, where relevant, to a specified code repository).

36 Taking these steps may require:

 (a) the inclusion of appropriate paragraphs in a notice of procurement (where applicable); and/or

 (b) the inclusion of specific contractual provisions in a draft contract; or

 (c) where an existing panel supply arrangement is being used, a review of the intellectual property provisions in the panel contract and, if required and possible, amendments to them for the purposes of the software development in question; and

 (d) ensuring that the contract does not include confidentiality provisions that would inadvertently prevent release of the software under an open source software licence.

## Act fairly towards developers when drafting IP warranties and indemnities

37 If:

 (a) a government agency is commissioning a service provider to develop software that is to be released on open source terms (either by the agency or by the service provider under a contractual obligation for the service provider to do so); and

 (b) it is known that the developer will be leveraging or adapting existing open source software developed by others,

the agency should act fairly towards the service provider in relation to the drafting of intellectual property warranties and indemnities.

38 In particular, it is generally considered unreasonable to expect a service provider to give an unqualified intellectual property (**IP**) warranty and an unqualified IP indemnity in relation to third party open source software that the agency agrees can be used by the service provider in developing software for the agency. This is especially so when the agency will be releasing the developed software under an open source software licence or requires the service provider to do so on its behalf. Sample IP indemnity and IP warranty clauses are provided in the Annexure to NZGOAL-SE.

## Liability

39 The MIT licence and the GPL both contain broad disclaimers of warranties and exclusions of liability that are widely known and acknowledged and ought to protect releasing agencies from liability in connection with the software they have released. In essence, people use free and open source software at their own risk. Agencies should ensure that all disclaimers and exclusions contained in the MIT licence or the GPL (as applicable) are replicated when they release software under these licences.

## Review and Release Process

40 Government agencies should follow the NZGOAL-SE Review and Release Process before publicly releasing and licensing their software for re-use under a free and open source software licence. The Review and Release Process is set out below.

# NZGOAL-SE Review and Release Process

## Introduction

41    It is recommended that government agencies follow the review and release process set out below before releasing software source code for re-use under a FOSS licence, with assistance where required from their technical and legal teams. The process consists of five main stages:

(a)    copyright-related rights evaluation;

(b)    evaluation of exceptions;

(c)    select a FOSS licence;

(d)    application of the chosen licence; and

(e)    release of the software.

42    Each stage contains one or more issues that may need to be worked through. The stages and the issues within them reflect a mixture of the NZGOAL-SE Policy Principles, legal requirements and practical considerations.

43    It can be important to work through these steps to ensure that the agency:

(a)    has all relevant rights in or to the software source code it proposes to publicly release;

(b)    uses the appropriate open source licence; and

(c)    does not expose either itself or those who may re-use the released software to liability or related risk.

44    A decision tree diagram for the review and release process is set out at paragraph 82 below.

## Stage 1: Copyright-related rights evaluation

### What the stage involves

45    The first stage involves:

(a)    clearly identifying the boundaries of the software that the agency proposes to release (e.g., the software may comprise a range of files, all of which would need to be released together);

(b)    determining whether that software constitutes or contains one or more copyright works; and, if so

(c)    evaluating whether the agency has sufficient rights to license the software under open source terms.

46    In the vast majority of cases, software that an agency wishes to license will constitute or contain one or more copyright works. In the unlikely event that a given piece of software does not constitute or contain one or more copyright works, an agency could, if no exception in Stage 2 applies, release it into the public domain using an NZGOAL-style "no known rights" statement. See NZGOAL for that statement. It is not discussed further in NZGOAL-SE.

### Issues where agency does not own all copyright

47    When an agency is in the situation of not owning all copyright in the software it would like to release and license for re-use and needs permission from the copyright owner(s), it is important to appreciate that the software may:

(a)    be all new source code (i.e., be a completely new copyright work);[16] or

(b)    build on pre-existing source code (i.e., be an adaptation / derivative of pre-existing source code).

48    The analysis for these two scenarios is different:

---

16    For example, the software could be a deliverable under a services contract the agency has with a vendor, but the contract may confer copyright ownership on the service provider and only license the software to the agency.

(a) **All new source code**: In the case of all new source code that the agency doesn't own but wishes to license, the agency would need to either:

(i) already be entitled, under a FOSS licence that the owner has recently applied to the software, to sub-license the software; or

(ii) obtain irrevocable written permission from the copyright owner to sub-license the software on the relevant open source terms.

**Notes on sub-licensing**: As to the first option, it is important to note three points. First, if the software has already been licensed under a FOSS licence, there may be no need for the agency to sub-license it because anyone who gets the software has the rights granted by the licence that the owner has applied. Second, some licences prohibit sub-licensing. For example, you cannot sub-license GPL-licensed software. Third, if there is a right to sub-license, that right would need to be broad enough to allow sub-licensing under the particular FOSS licence that the agency wishes to apply to the software.

(b) **Adaptation / derivative of pre-existing code**: In the case of an adaptation / derivative of pre-existing source code, the agency would need to:

(i) be able to identify who owns the pre-existing source code (noting that there could be more than 1 owner/contributor) and who has made and owns the new copyright in the adaptation / derivative work; and

(ii) to the extent that the agency does not own the copyright in the adaptation / derivative work, have permission from the other copyright owner(s) for their parts of the overall new work to be licensed under the FOSS licence the agency wishes to use.

To understand this, one needs to appreciate that an adaptation / derivative work consists of property (copyright) owned by the original licensor(s) (let's call them A) plus new and separate property (copyright) over the new original parts of the adaptation / derivative work that are created by B. The adaptation / derivative work is a distinct copyright work in its own right but B doesn't obtain property rights that are greater than B's own contribution. As a United States court has put it, "[t]he aspects of a derivative work added by the derivative author are that author's property, but the element drawn from the pre-existing work remains on grant from the owner of the pre-existing work".[17]

Depending on the number of pre-existing owners, the licences (if any) under which they may have released their code and the licence the agency wishes to apply to the adaptation / derivative work, this can get complex very quickly. In cases of any complexity, agencies may need to seek expert legal advice.

**Common scenarios where an agency will not be able to license software for re-use**

49 There are certain scenarios in which, from a copyright perspective, an agency will not be able to license software either under *any* FOSS licence or under a *particular* FOSS licence. Three common scenarios are as follows:

(a) **third party owner**: copyright in the software is owned by a third party and the agency is not permitted to license the software for re-use under a FOSS licence. In this scenario the agency cannot license the software under any FOSS licence;

(b) **adaptations of proprietary software**: the software is an adaptation or derivative of proprietary software and the agency does not have the proprietary software owner's written and irrevocable authorisation to license the adaptation or derivative under a FOSS licence (whilst the agency might own the copyright in its new contributions, it still needs authorisation from the owner of the base software that it adapted). In this scenario the agency cannot license the software under any FOSS licence; and

(c) **adaptations of GPL or similarly FOSS-licensed software**: the agency wishes to license the software under the MIT licence but the software is an adaptation or derivative of open source software that is licensed under a sharealike FOSS licence, such as the GPL, that requires distributed or conveyed adaptations or derivative works to be licensed under the

---

17    Stewart v Abend 495 U.S. 207, 223 (1990).

same licence. In this scenario the agency cannot license its adaptation under the MIT licence.

## Stage 2: Evaluation of exceptions

50    If an agency has completed Stage 1 and concluded that it does have sufficient rights to license the software under a FOSS licence, then the NZGOAL-SE Policy Principles recommend that the software be licensed under open source terms unless an exception set out in paragraph 18 applies.

51    For each proposed release, the exceptions need to be considered in the light of all the surrounding circumstances relevant to the specific software and its release.

52    In many instances, the exercise will be quick as none of the restrictions will apply. In that event, the agency can proceed to Stage 3 (Select a FOSS Licence) below. This is because the Open Access and Licensing Principle will not have been displaced.

53    If one or more of the exceptions applies, the Open Access and Licensing Principle will be displaced and not apply. One of two consequences will follow:

   (a)   **No FOSS licensing**: FOSS licensing of the source code may not be possible unless the reason for the exception(s) can be removed in accordance with paragraph 53(b). If FOSS licensing is not possible, the analysis ends and there is no need to consider the subsequent stages below.

   (b)   **FOSS licensing after amending or anonymising the source code**: If the source code can be amended or altered to remove the reason(s) for the exception(s) applying, the Open Access and Licensing Principle is resurrected and one progresses to Stage 3.

## Stage 3: Select a FOSS licence

### Introduction

54    Once an agency has determined that it has:

   (a)   the copyright-related rights it needs to license the software (in accordance with the Rights Clearance Principle); and

   (b)   none of the exceptions listed in paragraph 18 that would negate the Open Access and Licensing Principle applies,

   the agency can work through Stage 3 to select the appropriate FOSS licence before applying the selected licence in accordance with Stage 4.

55    In most cases, an agency's selection of a FOSS licence is likely to turn on the answers to one or more of the following questions:

   (a)   whether the source code builds upon and/or interacts with existing FOSS-licensed source code;

   (b)   if the source code does not build upon and/or interact with existing FOSS-licensed source code, whether a person who adapts the code and distributes it to others should be required to license the adaptations for re-use by others;

   (c)   whether there are known reasons or use cases for releasing the source code under a FOSS licence other than the recommended GPL or MIT licences.

56    NZGOAL-SE does recommend a default set of licences (GPL and MIT) but a "yes" answer to the first question may require (legally or ethically) the selection of a different FOSS licence and a "yes" answer to the third question might suggest that a more targeted FOSS licence is preferable. Each of the three questions is now discussed in turn.

### Source code builds upon and/or interacts with existing third party FOSS-licensed source code - respect existing licence(s)

57    If the source code builds upon and/or interacts with existing third party FOSS-licensed source code, the Respect Existing Licences Principle applies. In this scenario, an agency should – where possible – select the same licence(s) for its new software code as the licences applying to

the pre-existing source code (the upstream licences), or licences that are compatible with the upstream licence(s). For consistency reasons, using the same licence(s) is usually preferable to using compatible licence(s).

58    To give an example, if an agency developed a software library that interacts with a pre-existing open source application licensed under the Berkley Software Distribution (BSD) licence[18], it would be consistent with the Respect Existing Licences Principle for the agency to license its new source code under the BSD as well. Effectively, the BSD licence would become the selected FOSS licence as an outcome of the Stage 3 analysis.

59    Note, however, that the words "where possible" are used above because it will not always be possible to apply the same licences as the upstream licences. This can be the case where different parts of the pre-existing software the agency has used are licensed under different FOSS licences. For example, some of the pre-existing code might be licensed under the MIT licence (which is not a sharealike / copyleft licence) and other parts of the code might be licensed under the GPL (which is a sharealike / copyleft licence).

60    Where an agency uses these differently licensed parts of software in a broader project (e.g., the agency might adapt the pre-existing software) the agency may be required, by the sharealike licence, to release its new code under the sharealike licence when the agency distributes its new code. In other words, a sharealike licence like the GPL may trump the non-sharealike licence (in this example, the MIT licence). Generally speaking, if an agency is using multiple pieces of software in a project and those pieces are licensed under different FOSS licences (non-sharealike and sharealike), new code should be released under the sharealike licence and not the non-sharealike licence. See the Adaptations Principle in paragraphs 19-21 for further explanation and, if in doubt, seek legal advice before selecting and applying a FOSS licence.

61    To give an example, if an agency develops and distributes a plugin for a content management system that, legally speaking, is an adaptation of MIT-licensed code and GPL-licensed code that the agency has mixed together and used, the agency may need to license the plugin under the GPL. Whether there is an "adaptation" can be a difficult legal question in some cases but, if there is an adaptation, the GPL will trump.

**Entirely new source code - default FOSS licences**

62    As noted above, NZGOAL-SE recommends a default set of licences - two licences - for cases in which an agency wishes to release entirely new source code for re-use; in other words, for cases in which the source code to be released does not build upon and/or interact with existing FOSS-licensed source code.

63    The default licences, in no particular order, are the GPL licence and the MIT licence. These default licences and the reasons for selecting them as the NZGOAL-SE defaults are described in the NZGOAL Policy Principles (Open access and public release of agency software using free and open source licences).

64    NZGOAL-SE suggests that an agency's choice between these two default licences should turn on the answer to this simple question: would you like everyone to be able to re-use other people's distributed adaptations of the source code in the future?[19]

65    If the answer to this question is yes – that is, a person who adapts the code and distributes it to others should be required to license the adaptations for re-use by others – then the agency should select the GPL (version 3).

66    If the answer to this question is no – a person who adapts the code and distributes it to others does not need to license the adaptations for re-use by others – then the agency should select the MIT licence.

67    Answering the question yes or no is a judgement call for the agency.

68    In either case, the agency should proceed to Stage 4 (Application of chosen licence) unless there

---

18    See https://opensource.org/licenses/BSD-3-Clause
19    Agencies should note that, even if an agency decides to apply the GPL to its source code, the agency can always allow particular developers to use it under different terms if the agency wishes (as long as the agency is the copyright owner). It is conceivable, for example, that a New Zealand-based software developer may wish to use the code in proprietary software for which it has a strong export market. In that sort of case, the agency can make the code available to the developer under alternative licence terms, e.g. without an obligation to share the source code of distributed adaptations.

are known reasons or use cases for releasing the source code under a FOSS licence other than the recommended GPL or MIT licences (as discussed immediately below).

**Known reasons or use cases for releasing source code under an alternative FOSS licence**

69    In some cases an agency may wish to release entirely new source code and there may be specific known reasons or use cases for not opting for the GPL or MIT licences. Two examples of this are as follows:

(a)    **Libraries**: the source code to be released is a library of code and the agency knows that there are proprietary software suppliers who wish to use the agency's particular library within, or to link the library to, their own proprietary source code without having to release (or it being argued that they have to release) their proprietary code if they distribute their software. The GNU Lesser General Public Licence (LGPL) will allow this kind of use without exposing the downstream user to risk. The LGPL requires software licensed under it to be modifiable by end users through source code availability and does not require proprietary code that is linked to the LGPL-licensed library to be made available under the GPL.

(b)    **Server side use of licensed software**: the sharealike obligation in the GPL (i.e., the obligation to license one's adaptations under the GPL) is triggered upon distribution of an adaptation. Where software is installed on a server to provide a service to end users (e.g., in an application service provider or cloud services context), the sharealike obligation is not triggered. To some people this seemed like a loophole. The GNU Affero General Public Licence (AGPL) was designed to close that loophole. It is based on the GPL and sates that if source code is adapted then the "public use of a modified version, on a publicly accessible server, gives the public access to the source code of the modified version".[20] There may be cases in which an agency is FOSS-licensing software that is used on servers where the agency would like to achieve this result or where communities of interest ask the agency to achieve this result. If so, the agency may wish to select the GNU Affero General Public Licence (AGPL) instead of the GPL.

70    The selection of licences for unusual use cases like these can be difficult and in some cases using one of the NZGOAL-SE default licences (the GPL or MIT licence) may suffice. If in doubt, seek technical or legal advice in the context of your particular use case.

**Licensing of accompanying documentation**

71    Explanatory documentation files included alongside open source software code are usually separate copyright works. These works are often thought to be licenced under the FOSS licence that has been applied to the source code. However, most FOSS licences used for source code are not designed for documentation and often don't naturally or comfortably apply. A more appropriate approach is to apply a Creative Commons licence to documentation in accordance with NZGOAL.

72    Two situations need to be considered: the first is where the agency has created completely new documentation, the second where the agency has adapted pre-existing documentation.

(a)    **New documentation**: In the case of new documentation that the agency owns (i.e., it owns the copyright), the Creative Commons licensing is straight-forward. In this situation, NZGOAL-SE recommends that the Creative Commons licence choice reflect the nature of the selected FOSS licence. This means that an agency releasing new documentation alongside source code released under:

(i)    a *more permissive licence* such as the MIT licence, should license the documentation under a **Creative Commons Attribution 4.0 International licence**; and

(ii)    a *sharealike/copyleft licence* such as the GPL, should license the documentation under a **Creative Commons Attribution Sharealike 4.0 International licence**.

(b)    **Adapted documentation**: If an agency is adapting pre-existing documentation that accompanied pre-existing FOSS-licensed software, it may be necessary to respect the

---

20    See AGPL Preamble http://www.gnu.org/licenses/agpl.html

licence (if any) that had been applied to the pre-existing documentation. For example, if the pre-existing documentation had been licensed under a Creative Commons Attribution Sharealike (CC-BY-SA) 4.0 International licence then, when the agency publishes its adaptation, it would need to license the adaptation under a CC-BY-SA licence.

Sometimes an agency may find that existing documentation is licensed under an unfamiliar or unnecessarily complex licence. The agency may still wish to release its own documentation relating to, for example, its adapted part of the software, but the agency would prefer to use a Creative Commons licence for it. In this situation the agency may be able to create a new and separate document that contains the additional information the agency wishes to provide. That in all likelihood would be a new and standalone copyright work rather than an adaptation of an existing work. In that case, and assuming the agency owns the copyright in the new document, it could select a Creative Commons licence for the document.

## Stage 4: Application of chosen licence[21]

### Introduction

73     Stage 4 explains how agencies apply their chosen open source software licence to the software they're releasing.

74     Copyright and licensing statements for FOSS software generally comprise:

(a)     a statement of copyright ownership; and

(b)     a description of the FOSS licence that applies to the software.

They appear in the same place but are considered separately here to explain certain points.

### The copyright notice

75     When applying a licence, it is common practice to add the year the software was completed and the name of the copyright owner(s), in the following format:

```
Copyright (c) <year> <copyright holders>
```

76     If the licensing agency is a department of the Crown, it should also replace "Copyright" with "Crown copyright". For example, if the licensing agency were a department, its licensing statement would look something like this:

```
Crown copyright (c) 2015, Land Information New Zealand on behalf of the
New Zealand Government.
```

77     If the licensing agency is not part of the Crown (e.g., it might be a Crown entity), it would use the term "Copyright" rather than "Crown copyright" and it would state its name without reference to the "New Zealand Government". For example:

```
Copyright (c) 2015, Commerce Commission (New Zealand).
```

### Description of the FOSS licence that applies

78     The description of the FOSS licence that applies to the software either sets out the full terms of the FOSS licence or refers to the applicable licence and links to full text of the licence. The description may also contain statements that exclude warranties and liabilities. Examples can be seen in the next part of this Stage 4 below.

### Applying the selected licence

79     Agencies can apply their selected licence in one of two ways:

---

21     For a useful general discussion of this topic, see the Software Freedom Law Center's "Managing copyright information within a free software project" at http://softwarefreedom.org/resources/2012/ManagingCopyrightInformation.html

(a)   They can reproduce the full text of the licence, or a summary with a link to the full licence, at the top of each source code file they are licensing. Here are two examples:

**Example 1:** if the licensing agency were a department and had selected the MIT licence, its licensing statement at the top of each source code file would look like this:

```
MIT Licence

Crown copyright (c) 2015, Land Information New Zealand on behalf of
the New Zealand Government.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

**Example 2:** if the licensing agency were a Crown entity and had selected the GPL[22] licence, its licensing statement at the top of each source code file would look like this:

```
Copyright (c) 2015, Commerce Commission (New Zealand).

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or(at
your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <http://www.gnu.org/licenses/>
```

(b)   Alternatively, agencies can provide the licence as a separate text file (which would contain the full licence text) that accompanies the software files, usually in a top-level folder. If agencies take this approach:

(i)      the text file containing all the licensing text should be called "LICENCE" (or something similar); and

---

22    See "How to use GNU licenses for your own software" at http://www.gnu.org/licenses/gpl-howto.html

(ii)     the first paragraph of the licensing text that identifies the licensor should also identify the software to which the licence is being applied, e.g.:

```
SpatialZone Project, Crown copyright (c) 2015, Land Information New
Zealand on behalf of the New Zealand Government.
```

Note that, under this approach, if a single software file is separated from the distribution, the recipient is unlikely to see the applicable copyright notice. If this is of concern to an agency, the solution is to include a brief copyright notice in each file's header that points to the top-level LICENSING file. For example:

```
SpatialZone Project, Crown copyright (c) 2015, Land Information New
Zealand on behalf of the New Zealand Government. This file is
released under the <LICENCE> licence. See the LICENCE file found in
the top-level directory of this distribution for more information.
```

### Other licences

80    If for some reason you are applying an alternative open source software licence, you may wish to check whether the entity that maintains the licence has instructions on how to apply it. Alternatively, you could follow the approach suggested above for the MIT licence or the Free Software Foundation's suggested approach for the GPL.

### Stage 5: Release the software

81    Once the chosen licence has been applied to the software, release the software for re-use into one or more relevant code repositories and/or on the agency's website. Agencies are encouraged to:

(a)    use version control and existing source code repositories;

(b)    ensure the source code is publicly accessible;

(c)    use freely available tools for others to interact with the released source code;

(d)    include a CONTRIBUTING file to help others contribute to the source code and improve it as explained in the Contributions Principle;

(e)    include accompanying documentation such as a README file to help others get started re-using the software; and

(f)    consider whether to:

(i)    issue a press release about the agency's release of the software; and/or

(ii)    notify appropriate communities of interest.

### Review and release process decision tree

82    The decision tree diagram below illustrates the Review and Release Process explained above. It is intended to be read in conjunction with the explanations above for each stage.

**NZGOAL-SE Review and Release Process - Decision tree**

## 1. Copyright-related rights evaluation

Does the source code that the agency proposes to release constitute or contain one or more copyright works?
→ **No** → Consider releasing under NZGOAL-style 'no known rights' statement if no exception applies
→ **Yes** →

Does the agency own copyright in the work(s) or otherwise have sufficient rights to license the work(s)?
→ **No** → Do not license the source code for re-use
→ **Yes** →

## 2. Evaluation of exceptions

Are there any exceptions in the Review and Release Process that cannot be addressed by amendment or anonymisation before licensing the source code?
→ **Yes** → Do not release the source code under FOSS licence
→ **No** → **Proceed with release** (amend or anonymise the source code if required to address any exception).

## 3. Select FOSS licence

Does the source code build upon and/or interact with existing FOSS licensed source code?
→ **No** → Would you like everyone to be able to re-use other people's distributed adaptations of the source code in the future?
→ **Yes** →

**Respect existing licence(s)**
- Where possible, use same licence(s) as upstream licence(s) or compatible licence(s)*
- For sharealike/copyleft licensed source code, license documentation with CC-BY-SA 4.0 International licence**
- For more permissively licensed source code, license documentation with CC-BY 4.0 International licence**
- If in doubt, check with your legal team

Would you like everyone to be able to re-use other people's distributed adaptations of the source code in the future?
→ **Yes** →

**Use GPL licence**
- Sharealike/copyleft licence style
- Requires adaptations to be GPL-licensed if distributed
- License documentation with CC-BY-SA 4.0 International licence

→ **No** →

**Use MIT licence**
- More permissive licence style
- There is no obligation to license adaptations on distribution
- License documentation with CC-BY 4.0 International licence

Do you have any particular known use cases for releasing the source code under a FOSS license other than GPL or MIT?
→ **Yes** → Consider the other FOSS licence options (e.g. LGPL, AGPL). Check more complex use cases with your legal team.
→ **No** →

## 4. Application of chosen licence

Apply the selected FOSS licence in accordance with the instructions in Stage 4 of the Review and Release Process set out above.

## 5. Release the software

Release the software for re-use into one or more relevant source code repositories and/or on the agency's website. To the greatest extend practical, use version control and existing code repository platforms. Consider whether to (i) issue a press release about your release of the software and/or (ii) notify appropriate communities of interest.

**Notes**

* Using more permissively-licensed source code (e.g., MIT-licensed source code) and sharealike/copyleft-licensed source code (e.g., GPL-licensed source code) together in a broader project may require you to release your new source code under the same sharealike/copyleft licence or a compatible licence rather than under the more permissive licence. There is potential complexity here so it's advisable to consult your legal team for guidance.
** If you're amending or updating documentation that is licensed under a different licence, you may need to license your adaptations under the same licence or a compatible licence. Again, consult your legal team for guidance.

# Annexure: Specimen IP warranty and IP indemnity clauses

## Introduction

1    The specimen IP warranty and IP indemnity clauses set out below are starting points for agencies that wish to act in accordance with the "Act fairly towards developers when drafting IP warranties and indemnities" Policy Principle in NZGOAL-SE. They are designed for insertion into new contracts for the development of software where FOSS software might be used. They can also be used to amend the standard terms of the Government Model Contract for Services (which is done by making appropriate insertions into Schedule 2 of that Contract).

2    The precise drafting of such clauses in any given case, in the context of an agency's project and the contract it is using, is a matter for the agency. The provision of these specimen clauses does not constitute legal advice to any agency or person.

## Context

3    It is common for software development contracts to contain a warranty clause along these lines:

> **1.  Service Provider Warranties**
>
> The Service Provider represents and warrants that the possession or use by the Customer or its Personnel of any item of Intellectual Property supplied or licensed by the Service Provider will not infringe the Intellectual Property rights of any third party.

4    It is also common for software development contracts to contain IP indemnity provisions that include a clause like this (this is just an example; sometimes they are more elaborate):

> **2.  IP Indemnity**
>
> (a)    The Service Provider will indemnify the Customer against all Losses suffered or incurred by the Customer or its Personnel as a result of any claim that the possession or use by the Customer or its Personnel of the System or any other Intellectual Property or materials supplied or licensed by the Service Provider under this Agreement infringes any third party's Intellectual Property rights.

5    The specimen clauses set out below are based on the assumption that there are such clauses in the development contract in question.

## Specimen clauses

### Specimen open source software clause

6    The specimen open source software clause set out below:

(a)    requires the service provider to obtain the customer's consent before using particular open source software (this is not due to any judgement on the use of open source software but to enable the customer to be aware of the use of open source software and to assess the applicable FOSS licence(s) if it wishes);

(b)    gives primacy to the FOSS licence(s) used in relation to the software they cover;

(c)    confers ownership of new software IP on the customer;

(d)    qualifies the application of the service provider warranty mentioned above to open source software;

(e)    requires the service provider to comply with applicable FOSS licences; and

(f)    requires the service provider to release adapted FOSS software or new software designed to interact with FOSS software to an open source repository if the agency requests it to do so (this does not prevent the agency from doing so but some agencies may prefer their service providers to do this).

7    This is the specimen clause:

**3.    Open Source Software**

(a)    The Service Provider shall obtain the Customer's written consent before providing, incorporating or adapting, or developing any widget, plugin or module for, any Open Source Software, in the course of its performance and delivering the Services and any associated Deliverables. The parties acknowledge that the purpose of this requirement is to enable the Customer to review the applicable open source licence terms, make a decision as to whether those terms are acceptable and, should consent be granted, respect any obligations in those licence terms.

(b)    If the Customer grants such consent:

(i)    the terms of the applicable open source licence(s) will apply to that Open Source Software;

(ii)    the provisions of that licence or those licences will prevail over the terms of this Agreement in the event and to the extent of any inconsistency;

(iii)    all Intellectual Property rights in the Open Source Software that is supplied to the Customer (or incorporated into the System or any other Deliverables supplied to the Customer) under this Agreement, or under any collateral licence or agreement, remains the sole and exclusive property of the relevant third party;

(iv)    the Service Provider will own the new Intellectual Property rights in any adaptation or derivative of the Open Source Software and in any widget, plugin or module the Service Provider develops under this Agreement that is designed to interact with the Open Source Software;

(v)    the warranty in clause 1 does not apply in relation to that Open Source Software unless the Service Provider knows or has reason to believe, at the time it selects the Open Source Software, that the Open Source Software would, or would be likely to, infringe the Intellectual Property rights of a third party;

(vi)    the Service Provider will comply with all applicable terms of the open source licence(s); and

(vii)    if required by the Customer, the Service Provider will release:

(A)    any adaptation or derivative of the Open Source Software provided or incorporated into the System; and

(B)    any widget, plugin or module the Service Provider develops under this Agreement that is designed to interact with the Open Source Software and is used in the course of its performance and delivering the Services and any associated Deliverables,

to one or more specified online software code repositories and under the open source licence that is either:

(C)    required to be applied to the adaptation, derivative, widget, plugin or module by the upstream open source licence; or, if there is no such requirement

(D)    specified by the Customer,

provided that the Customer first consults the Service Provider for a minimum of 5 business days before exercising its right in this clause 3.2(b)(vii).

**Specimen modification to IP indemnity clause**

8    Sub-clause 2(b) in the specimen clause below has the effect of limiting the application of the IP indemnity mentioned above to open source software that the Service Provider uses in providing services and deliverables to the customer. It would usually form part of the indemnity clause mentioned above. For this reason, that indemnity clause is repeated here:

**2.    IP Indemnity**

(a)    The Service Provider will indemnify the Customer against all Losses suffered or incurred by the Customer or its Personnel as a result of any claim that the possession or use by the Customer or its Personnel of the System or any other Intellectual Property or materials supplied or licensed by the Service Provider under this Agreement infringes any third party's Intellectual Property rights.

(b)    Clause 2(a) does not apply to any Losses arising from the Customer's possession or use of any Open Source Software, unless the Service Provider knew or believed, at the time of selecting the relevant Open Source Software, that the possession or use by the Customer or its Personnel of that Open Source Software would, or would be likely to, infringe any third party's Intellectual Property rights.

**Associated definitions**

9    The specimen clauses above use various defined terms (these are the terms that are capitalised). Not all of the defined terms need to be set out here. The relevant defined term for present purposes is the definition of Open Source Software. There are various ways to define Open Source Software. This is one way:

**Open Source Software** means any software that is released on licence terms that grant others the freedom to use, copy, modify and distribute the software, for either non-commercial or commercial purposes, as long as they comply with the applicable licensing conditions.