# How to FOSS Your Government Project Version 1

TEMPLATE

This document is intended as a template for an agency policy/checklist for how to release government-developed software as Free and Open Source Software (FOSS). You can use this document as a generic process for your specific project, or as a template for your agency or department.

If you are using this as a template, sections highlighted in YELLOW/ALL CAPS are intended to be replaced by localized instructions specific to the processes of your organization. Delete or replace the highlighted items.

Comments or suggestions on how to improve this checklist should be sent to Daniel.Risacher@osd.mil.

# 1 Introduction:

This document is an aid to AGENCY-sponsored software development projects moving to a public, unclassified, Internet-accessible open source environment. It focuses on the considerations, policies, and processes that apply to the transition. It does not delve into best practices for running an open source project as there are numerous resources on that topic. (One of the best is Karl Fogel's free book, Producing Open Source Software. How to Run a Successful Free Software Project, available at [www.producingoss.com](http://www.producingoss.com).)

The document begins by presenting the many concerns a project team must address before making the move and offers some advice where appropriate. The document ends with a checklist that summarizes the tasks to be completed in order to legally and successfully transition to a public environment.

Before beginning the transition, the project and management teams need to weigh the costs of moving to a public open source approach against the perceived gains.

* What is the business case for moving to open source?
  + How does it benefit the project, program, AGENCY, and/or DoD?
* Is releasing the source code the best option for achieving the goals?
  + Is releasing to the public the best option or is a government-only release of the source code a better approach?

Starting with the first discussion a project team has about moving to open source, it is important to keep good records of the activities including meetings, emails, decisions, reasoning, etc. These records will be valuable to future project members and managers as well as any process, procedure, or legal reviews.

# 2 Copyright

To release something to open source, you must have the legal right to do so. For the government, this typically occurs as one of the following scenarios:

* The work is in the public domain; therefore, there is no copyright issue. This is the default case for software created by government employees (civilians and military).
* The government holds "unlimited rights", which is often the situation for work created on behalf of or at the direction of the government. In this scenario, the creator of the work, usually the organization contracted to write the software, holds the copyright and the government is provided unlimited rights to the work via clauses in the contract. The most common clause with respect to software rights and the government is DFARS 252.227-7014.
* The government holds the copyright to the work. This is less often the case as it requires the contracted creator to assign total copyright to the government. A common contract clause for this is DFARS 252.227-7020.

Dr. David Wheeler wrote an excellent article on copyrights in relation to the government releasing software as open source. It includes some easy-to-read tables of scenarios that come about from the more common FAR and DFARS clauses used in government contracts. The article, "Publicly Releasing Open Source Software Developed for the U.S. Government", was published in The Journal of Software Technology and can be found at [www.journal.thedacs.com/issue/56/180](http://www.journal.thedacs.com/issue/56/180) . Refer to the article as background material for understanding what rights the government has to the software being considered for release.

Take special note of what rights are provided to the government in contracts and contributor agreements. Among them must be the right to sub-license and/or release under a different license depending on the project's release strategy.

#### Who Should Hold the Government's Rights?

Typically, when negotiating, establishing, and documenting rights for the government, they should be assigned to the "United States Government" unless there is a strong reason for a sub­organization to hold the rights.

#### Documenting the Government's Rights

Establishing the government's rights to release is accomplished and documented via contracts and contributor agreements. When creating or modifying a contract for activities to develop software for release to open source, work with the Contracting Officer (CO) and Contracting Officer Representative (COR) to select the appropriate FAR and DFARS clauses that will ensure the desired release-ability of all artifacts produced as part of the contracted work. Have the Office of General Counsel (OGC) review and approve the contract wording and inclusion of the FAR and DFARS clauses. For work and contributions not covered by a contract (for example voluntary contributions from the open source community) establish a contributor agreement that provides the necessary rights to the government. Have OGC review and approve this agreement. Have all contributors not otherwise covered under contract sign the agreement before making contributions to the project.

For contributions made before the appropriate documents were in place (contract clauses and/or contributor agreements), the government must either be provided with the necessary rights to license and publish the work or exclude the contribution from the release. Each copyright holder should assign the appropriate rights to the government by signing a document similar to the Contributor Agreement, but covering the works/contributions made previously. OGC should review and approve this agreement/document.

#### Government Contributions

Works of the United States government are not subject to copyright protection under the Copyright Act (17 USC §105)- as such they are "Public Domain" and not copyrightable.

#### Notices

Be sure to include proper copyright notices in all software code files, script files, documents, README file, etc. A copyright notice should also appear in a "COPYING" file at the top directory of a software tree and posted on project web pages.

Note that artifacts containing contributions from multiple sources should include notices for each of the copyright holders. Further, if contributions are provided under differing rights and/or licenses, separate notices should be included for each.

# 3 Patents

Holding the copyright or having similar rights assigned to the government does not eliminate patent concerns. It is possible for contributors to hold or later be assigned, a patent for an algorithm or approach implemented in software files to which the government has complete or shared copyright. Further note the right to use a patented work does not imply the right to allow others to use it.

Some FAR and DFARS clauses address patent rights, but many do not. Be sure the program's goals are met with respect to patents in all contracts.

Contributor agreements should include specific clauses that clarify patent awards, ownership, and rights.

#### Government-Held Patents

This document does not address issues of sharing information patented by the government. For advice on those issues, contact the Technology Transfer Office and the AGENCY Policy Office.

# 4 Trademarks

Protect the project's name, logos etc., by registering trademarks.

# 5 License

Which license an open source project uses depends on that project's goals and philosophies. For example a project's values and goals may include wide adoption, building an active development community, compelling community sharing of enhancements and derivative works, enabling proprietary knock-offs, preventing proprietary knock-offs, and others. The project needs to pick a license that serves its particular purposes.

If a project holds the necessary rights to license software, there are few restrictions on which license can be applied. However, when choosing an open source license, there are two widely accepted rules that should be observed:

1. Do not create a new license. Dozens of good ones exist already and they accommodate nearly every project situation. Also, the open source communities understand them and are comfortable with them.
2. Pick a license from the Open Source Initiative's (OSI's) "Approved licenses" list. See http://www.opensource.org for the list of approved licenses, descriptions of each license, an explanation of the approval process, and what it means for a license to be approved.

For a thorough introduction to open source licensing, refer to the OSS Watch (Open Source Software Advisory Service) web pages at http://www.oss-watch. ac.uk. They and other sources, such as the OSI at <http://www.opensource.org> and the Free Software Foundation (FSF) at <http://www.fsf.org>, offer advice on choosing licenses and license compatibility.

Be aware the project's license will affect the community's acceptance and adoption of the project in two major ways:

1. It legally denies and allows certain actions by those using or desiring to use the project's offerings. The rights provided by the license must align with their needs.
2. It, along with the project's governance model, indicates to the community the project's open source philosophies. The open source community can be fickle. Perceptions (right or wrong) can play a strong role in the decision to accept project code or join/contribute to a project.

#### License Compatibility

License compatibility is a concern in two areas:

1. If your software depends on any other software licensed to you, the license you choose for your project must be compatible with those licenses.
2. Your license will dictate what developers building on your code will be able to do and what license choices they have for their products.

#### Switching Licenses

If the copyrights or appropriate rights are held by the project or government, then a project can switch licenses at any time. However, be aware that previous versions of code remain available under their previous licenses. So moving to a more restrictive license could cause the community to fork the project under the less-restrictive license.

If 3rd party contributions to project have been included under license (but without a contributor agreement which grants the rights beyond the original license), then it may not be possible to change licenses.

#### Dual-Licensing

Dual or multi licensing schemes are useful to solve license compatibility issues and to provide market segregation based on business models.

#### Public Domain

Works generated by U.S. federal employees are public domain and should contain declarations thereof.

Dedicating an entire project as public domain can be done, but is often more complicated than expected and can lead to legal problems in the future. Usually, it is preferable to release the project under a liberal license that provides the same freedom as public domain, but does so explicitly and with more legal backing.

Often, the government does not have all the rights to all the components of a project necessary to dedicate it as public domain.

Further, there are some legal issues that make releasing to the public domain unclear in some situations. For example:

* Countries have different definitions of “public domain” especially works originating outside their borders.
* U.S. federal employees' products are only “unambiguously” free of copyright in the U.S.

Providing the products under a license also provides for the inclusion of disclaimers to avoid lawsuits for harm or damages, etc.

If the product is to be released to the public domain, it is good practice to do so using the Creative Commons' CC0 Public Domain Dedication that places the work into the public domain and provides a fallback license for cases where the public domain is in question. The fallback license attempts to grant all the rights to all users as though it were public domain.

#### Notices

The license should be distributed with the code base and links to the license file and/or its URL should be included in every file to which it applies (source code, document, script, configuration, etc.) and in the README, and COPYING files.

Note that artifacts containing contributions from multiple sources should include notices for each of the copyright holders. Further, if contributions are provided under differing rights and/or licenses, separate notices should be included for each.

In addition to license compatibility considerations, code or other types of artifacts from other projects may require attribution. Generally, the best way to handle these requirements is to include a NOTICE.txt file which states information about the bundled inclusions, their licenses, and any attribution statements. Licenses requiring attribution will typically be very specific about how it is to be accomplished.

Because licenses and license schemes can be complicated, it is best practice to post the project's intent with respect to licenses on its web page and in its documentation. For example, a project may want to copyleft its framework to ensure the community has a solid and shared infrastructure, but allow widgets and components that utilize the framework to be proprietary. This can be stated clearly on a web page but may not be obvious to someone reading the project's licenses.

# 6 Transferring Data from Classified Systems to Unclassified Systems

To transfer (move or copy) data from a classified system to an unclassified/unsecured system, the data-must be verified as unclassified, approved for release, and established procedures for accomplishing the physical transfer must be followed.

#### Establishing Data as Unclassified

INSERT AGENCY-SPECIFIC PROCEDURES HERE

#### Transferring Data

INSERT AGENCY-SPECIFIC PROCEDURES FOR HIGH-TO-LOW TRANSFERS HERE

# 7 Accepting Contributions

When a government project accepts contributions from the community, there are two main topic areas to consider. The first area is covered under the project's governance model. The model will specify the conditions and processes under which software contributions will be offered, reviewed and accepted. Discussions of various governance models can be found in books and the Internet. Again, for a more detailed discussion, I recommend Karl Fogel's book: Producing Open Source Software, How to Run a Successful Free Software Project, which can be found at [www.producingoss.com](http://www.producingoss.com).

The second area of concern is the process for bringing the software from unclassified/unsecured systems (i.e. the Internet) to classified/secure systems. Although a review of the contribution would have taken place on the unclassified system before accepting it to the project's code base, it is even more important to ensure a review by trusted and competent authorities (i.e. cleared developers from the project team) takes place before considering bringing the contribution to a protected system. Following the code review, proper security procedures must be followed for the physical movement of the data to the protected system. REFERENCE TO AGENCY-SPECIFIC PROCEDURES FOR LOW-TO-HIGH TRANSFERS HERE.

The review process and procedures should be documented as an attachment to the final approval documentation.

# 8 Prepublication/Public Release Process

“Pre-publication” is a the term used in the Intelligence Community(IC) for the process of reviewing information before it can be released to the public. For organizations outside the IC, the analogous term is usually the “Security/Policy Review” process. The term “Public Release” is also used.

#### 8.1 Initial Release

Before the software, documents, etc. can be released to the public, all artifacts must go through the process of AGENCY REVIEW OFFICE where they are reviewed and approved for release. Project representatives should meet with AGENCY REVIEW OFFICE to discuss the project, its move to open source and the types and quantity of documents, code files, etc. the project plans to release. AGENCY REVIEW OFFICE will explain the requirements and how they want to handle the artifact review. AGENCY REVIEW OFFICE may rely on the reviews of subject matter experts from the project office and approvals/concurrences from the project's supervisory chain (division chief or higher) to confirm the nature and classification of the collection of artifacts as a whole.

AGENCY REVIEW OFFICE does not review software (source code).The process for the release of source code is for the project office to forward the source code to AGENCY SPECIFIC PROCESS HERE.

* A digitally signed email from a technical person (other than the author) who reviewed the code and verified it as unclassified. (Looked for code words, server names, IP addresses, project names, etc.)
* A digitally signed email from the APPROPRIATE OFFICAL confirming the code as unclassified.
* A digitally signed email from the project's management (division or higher) stating the code is unclassified and it should be shared.

Alternatively, there is a DoD-wide process for public release review run by the Office of Security Review (OSR) in the Washington Headquarters Service (WHS). This process is available to DoD employees, retirees, and industry partners. See DoD Instruction 5230.29 “Security and Policy Review” and <http://www.dtic.mil/whs/esd/osr> for more information.

#### 8.2 Ongoing Process

Successful open source projects require good community communication. There are frequent code releases, email exchanges, and community forums that need quick responses and constant interaction. However, postings to public areas by project personnel are required to pass a prepublication review before being released. This requirement presents a problem for the fast­ paced environment of open source projects. One solution is to document how the project will interact with the open source community and have its management chain (and the prepublication office, for IC agencies) approve future interactions under the conditions specified in the document. The document should include the following:

* Description of subject areas that are approved for future code releases and discussion
* Description of what types of artifacts project personnel can release. For example, software files, project documents (user manuals, configuration manuals, etc.), email, forum postings.
* Description of the venues where project personnel are permitted to interact. For example, the code repository, project email lists, project forum.
* Process for determining if a questionable topic is covered in the pre-approval agreement

The project should also document and institute a process for training personnel on the approved community interaction procedures. This should include a review and discussion of the prepublication agreement described above and guidelines on how to deal with public inquiries outside the approved subject areas.

# 9 International Traffic in Arms Regulations (ITAR)

There are no actions to take in regard to ITAR other than to be aware of how the software is exempt from those regulations:

DoD does not have authority to grant export control licenses; however, 22 CFR part 125.4(b)(13) states that technical data is exempt from ITAR export controls if it is "approved for public release (i.e., unlimited distribution) by the cognizant U.S. government department or agency or Office of Freedom of Information and Security Review."

Similarly, the software, once approved for public release, is not subject to Export Administration Regulations (EAR). Ref 15 CFR part 734.3(b)(3).

For reference:

* LIST AGENCY POC for ITAR decisions, if known.
* LIST AGENCY POC for Export Control Policy, if known.

# 10 Approvals and Notifications

Stakeholders must be aware of and comfortable with the project's move to open source. Depending on the stakeholder, this can be accomplished via email, INTERNAL AGENCY PROCESS, documentation, and most often meetings and briefings. Some groups of stakeholders should attend meetings together to facilitate open discussion of plans and concerns. Some stakeholders must provide approval before the program can proceed to open source, others only require notification of the move.

#### 10.1 Program's chain of command

The program's leadership chain must support the move to open source. Typically they demonstrate their approval via email threads.

#### 10.2 Office of General Counsel (OGC)

The OGC can help identify the necessary FAR and DFARS clauses for contracts. They will also review the contracts, contributor agreements, etc. to ensure the government has the necessary rights to release the project artifacts to the public.

The OGC will also review the license(s) and licensing schemes under which the project will be releasing the software. They will determine if the government can legally release the software under the licenses recommended by the project.

#### 10.3 OTHER KEY AGENCY STAKEHOLDERS

There may be other key stakeholders within the organization who must evaluate equity issues with respect to the project's release to the public. Typically this occurs as a meeting between project representatives and directorate-level leadership. Document the decision with meeting minutes and/or email.

#### 10.4 Chief Information Officer (CIO) and Chief Technical Officer (CTO)

The AGENCY CIO and AGENCY CTO need to understand the project and the project's move to open source. The best venue is a meeting with them to discuss the project, the benefits of moving to a public open source environment, and what steps have been taken to get approval for the release. The final approval authority may ask the CIO and/or CTO for advice on the activity. They need to understand the project's goals and situation.

#### 10.5 Technology Transfer Office (TTO)

Discuss the program's release of software with the Technology Transfer Office. Although the TTO typically deals with patents and licensing where the agency/department expects to receive a return on investment by transferring technology to industry, they should be informed of the project's plan to release its software.

#### 10.6 Public Affairs Office (PAO)

The Public Affairs Office (PAO) needs to know what the project is, what it does, and information about its release including the license scheme, expected community involvement, and release date. It is likely they will receive questions about the project from the public and possible media so they need to be informed prior to the project's release.

#### 10.7 Final Approval

The release approval authority will likely be AGENCY-SPECIFIC; however, the approval may need to come from higher.

The approval will come in the form of AGENCY-SPECIFIC APPROVAL PACKAGE. The approval package should include the appropriate attachments (ex. Project's classification guide, prepublication approval for release, description of community interaction- agreed to by prepublication office, description of contribution review and ingest process) and the following sections:

* Purpose: State that the project wants approval to release to open source. Specify important aspects like the license
* Background: Relay important background information pertinent to why the project should be released like how the project has already been shared and accepted by other organizations
* Discussion: Explain what the project is/does and how the Agency and/or DoD would benefit from the release. Refer to attachments for prepublication approval for release, the prepublication agreement for future releases and community interaction (with the document that describes the process), and anything else relevant (for example how the project will ingest community contributions)
* Recommendation: Ask for approval to release

# 11 Establishing and Running the Open Source Project

How to run an open source project is a topic beyond the scope of this paper. There are hundreds of works available to guide a project through establishing its presence and producing software. Fogel's free book mentioned elsewhere in this paper is one of the great resources available on this topic.

As such, this paper will touch only on a few topics that need special consideration from government "run" projects.

#### 11.1 Communication

Often, government software projects are conducted by co-located teams who communicate among themselves via face-to-face meetings or point-to-point email. This approach is adequate for closed and pseudo-closed development, but should be avoided in open source projects.

Communication among the team should take place on the project's forum or via the project email lists. If an impromptu meeting occurs where a decision was made (no matter how small) or an important concept was discussed, the contents of that meeting should be posted to the forum. This serves several purposes: it documents important decisions, it documents important concepts, concerns, and approaches, it lets everyone in the community know what others are working on, it enforces equality within the community- i.e. members not working at site X are just as important as those team members who have the advantage of being co-located.

#### 11.2 Hosting and Distribution

Use a public hosting environment like SourceForge or GitHub to host the project. There are members of the public who are suspicious and doubtful of anything to do with the government­ especially free things coming from the government. Don't add to this by standing up a hosting platform/environment just for the project. Pick something the public already uses and trusts.

Additionally, there are huge cost savings that come from using a free, public hosting environment.

#### 11.3 Providing Verification and Security of Shared Files

Ensure the integrity of the project's offerings (code, documents, binaries, archives, etc.) by incorporating security and verification techniques like digital signatures and hash codes to all artifacts.

#### 11.4 Governance

When choosing the open source project's governance model, plan ahead for special government­ related circumstances. For example, knowing code contributions will likely be moved to government systems may necessitate all committers be cleared project members. These types of requirements may prevent the project from turning governance over to others like Apache.

#### 11.4.1 Accepting Contributions

There are special considerations for accepting contributions from the public if the project plans to use them on government systems. Refer to section 7 - Accepting Contributions.

#### 11.5 Press Release

When the project's open source web presence is ready, the project should make announcements in internal publications and external publications like Freshmeat. The external announcements should be reviewed by the Public Affairs Office.

**Acronyms and Abbreviations**

CFR Code of Federal Regulations

CIO Chief Information Officer

CO Contracting Officer

COR Contracting Officer’s Representative

CSS Central Security Service

DCSA Department of Defense (DoD) Community Source Agreement

DFARS Defense Federal Aquisiton Regulation Supplement

DoD Department of Defense

EAR Export Administration Regulation

FAR Federal Acquisition Regulation

FOSS Free & Open Source Software

FSF Free Software Foundation

GPL GNU General Public License

ISSM Information Systems Security Manger

ISSO Information Systems Security Officer

ITAR International Traffic in Arms Regulation

NDAA National Defense Authorization Act

OGC Office of General Counsel

OSI Open Source Initiative

OSS Open Source Software

PAO Public Affairs Office

SME Subject Matter Expert

TTO Technology Transfer Office

USC United States Code

**Establishing a Government Project as FOSS: Checklist**

## A. Copyright and Patents

□ Government entity selected to hold copyright: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#### Current and Future

□ FAR and/or DFARS clauses in place (contracts)

□ Contracting Officer (CO) and Contracting Officer Representative (COR) ensured the contracts support the desired outcome for release-ability of artifacts.

□ Office of General Counsel (OGC) has reviewed and approved wording of contracts and inclusion of FAR and DFARS clauses.

□ Current contributors signed Contributor Agreement (or covered under appropriate contract)

□ Contributor Agreement in place (for future contributors)

#### Past (contributions made before current contracts and contributor agreements)

□ Previous contributors have provided rights to project/government

#### Government Contributions

There is no action to take in regards to government contributions. This section is included for completeness.

Works of the United States government are not subject to copyright protection under the Copyright Act (17 USC §105) -as such they are "Public Domain" and not copyrightable.

#### Notices

□ Copyright notice included in all code files,documents,README file,and COPYING file

□ Public Domain dedication or declaration statement for works in the public domain

□ Copyright notice posted on project web pages

## B. License

□ License selected: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

□ Office of General Counsel (OGC) has reviewed and approved license

□ If including code or other artifacts to which the government does not hold the rights, license compatibility has been verified

□ license text included in LICENSE.txt file and COPYING file

□ Reference to license (name, URL and/or LICENSE file) in code files and documents

□Required attributions, license notices, and bundled library lists in NOTICE.txt file.

□ License text posted on web page.

□ Project's intent, with respect to licensing, posted on web page and in documentation.

## C. Trademarks

□ Trademarks registered for logos,names,phrases,art, etc.

## D. Transferring Data from Classified Systems to Unclassified Systems

#### i. Establishing Data as Unclassified

□ Classification of artifacts have been reviewed by appropriate offices.

□ Prepublication/Public Release office has reviewed and approved all artifacts for release/publication. (Refer to F – Prepublication/Public Release Process)

#### ii. Transferring Data

□ Procedures have been established for moving the artifacts from the classified system to an unclassified system in accordance with agency policy (if necessary)

□ Procedures have been reviewed and approved by Information System Security Manager (ISSM)

□ Procedures have been reviewed and approved by Information System Security Officer (ISSO)

## E. Accepting Contributions

□ Procedures for contribution review and movement to classified system documented if necessary.

## F. Prepublication or Security/Policy Review

#### i. Initial Release

□ Assemble artifacts required for review:

* A copy of the source code
* A digitally signed email from a technical person (other than the author) who reviewed the code and verified it as unclassified. (Looked for code words, server names, IP addresses, project names, etc.)
* A digitally signed email from the appropriate office confirming the code as unclassified.
* A digitally signed email from the project's management (division or higher) stating the code is unclassified and it should be shared.

□ Prepublication or Security/Policy Review process has reviewed and approved all artifacts for release/publication

#### ii. Continuing Activities

□ Open source community engagement processes and procedures have been documented and agreed to by the program's management chain (and the prepublication office, if needed). Documented with email threads.

□ Project has established processes for educating team members on approved community interaction policy and procedures.

## G. International Traffic in Arms Regulations (ITAR)

There are no actions to take in regards to ITAR. This section is included for completeness.

22 CFR part 125.4 (b) (13) notes that when data is approved for public release it is exempt from ITAR export controls.

## H. Approvals and Notifications

#### i. Program's chain of command

□ Division Chief and Office Chief concur with activity to move project to open source. Documented with email threads.

#### ii. Other Stakeholders

□ OTHER-AGENCY STAKEHOLDER concurs with activity. Documented with email threads or meeting minutes.

#### iii. Chief Information Officer (CIO) and Chief Technology Officer (CTO)

□ Agency CIO and CTO approve activity. Documented with email thread or meeting minutes.

#### iv. Technology Transfer Office (TTO)

□ TTO has been notified of activity. Documented with email thread or meeting minutes.

#### v. Public Affairs Office (PAO)

□ PAO has been notified of activity. Documented with email thread or meeting minutes.

#### vi. Office of General Counsel (OGC)

□ OGC has reviewed the contracts (including FAR and DFARS clauses),contributor agreements, etc. and concurs that the government has the necessary rights to release the artifacts to the public. Document with meeting minutes and/or email threads

□ OGC has reviewed the license(s) and licensing schemes under which the software is being released and confirms the government is acting legally. Document with meeting minutes and/or email threads.

#### vii. Final Approval

□ Documented approval

Required approval attachments:

* Release guide- describes project ongoing interaction with community
  + Approval/agreement notice from prepublication
* Security/Policy review approval
* Description of open source project governance

Optional attachments (depending on project):

* Classification guide
* List of activities leading up to release approval

## I. Protecting Offerings

□ Artifacts (source code files, archives, binaries, documents, etc.) protected with appropriate security measures (ex. Digital signatures, hashes).