STATE OF OPENNESS IN INDIA's E-GOVERNANCE APPLICATIONS

March 25, 2025

By UPASANA HEMBRAM

REVIEWED BY- DIVYANSHA SEHGAL

Supported By- Mozilla Foundation

State of Openness in India's E-Governance Applications

1. Introduction

Open source software (OSS), also commonly known as free and open source software (FOSS) or free libre open source software (FLOSS), is software that is made available with its source code. It is licensed liberally, granting users access to study, use, modify, improve, or redistribute it.[1] In this context, the term 'open' refers to the source code being made available without having to pay royalties or licensing fees, while the term 'free' refers to the freedom to copy and use the software rather than being 'free of cost'. The two organisations that are the self-appointed custodians of these definitions are the Free Software Foundation (FSF)[2] and the Open Software Initiative (OSI).[3] While the two organisations and the two terms resulted from different philosophies and represent different methodologies, the FSF and OSI acknowledge that for all practical purposes, "they both refer to essentially the same thing"[4]; "however, the differences in extension of the category are small: nearly all free software is open source, and nearly all open source software is free."[5]

FOSS is an inclusive term that covers free software, and open source software and is used without any political, philosophical, or cultural bias towards either of the approaches. In most cases, free software licences and open source licences are the same. The combined philosophical underpinning of the FOSS movement is the belief that software should be free and open for everyone to use and learn from.[6]

Often, FOSS is mistaken for simply releasing the software source code publicly. Providing access to the source code is a necessary but not sufficient condition for the software to be free and open. The FSF specifically focuses on the philosophical freedoms of users, and OSI lays down 10 criteria for the distribution of software that makes it open source.[7] These robust criteria

[1] "Open Source for Global Public Goods", World Bank, 2019, https://documents1.worldbank.org/curated/en/672901582561140400/pdf/Open-Source-for-Global-Public-Goods.pdf
[2] https://www.fsf.org/
[3] https://opensource.org/
[4] https://www.debian.org/intro/free
[5] https://www.gnu.org/philosophy/categories.html
[6] https://www.digitalocean.com/community/tutorials/free-vs-open-source-software
[7] https://opensource.org/osd/

include the distribution of the software along with its source code under an open licence that allows for broad use of the software; the source code is also accompanied by documentation and guidance so that it can be used by everyone without any restrictions or discrimination. As evident in the language of various Government of India (GoI) policies and guidelines, the GoI also abides by this definition.[8]

In the post-COVID-19 world, digitisation has become imperative, with governments across the world increasingly digitising their services to serve their citizens remotely. Against this background, the widespread adoption of FOSS is being recognised as a powerful approach to accelerate digitisation efforts by multilateral organisations, funders, and even governments.[9] However, despite the increasing momentum and the recognition of the potential benefits of FOSS by several departments of the union government, government departments are still locked in contracts with proprietary software vendors. Despite the GoI's pro-FOSS stance, a few of its initiatives have drawn criticism for open washing and not actively engaging with the Indian FOSS community.

2. Methodology

We undertook an extensive literature review of what openness stands for, based on the principles, philosophies, and discussions outlined by the OSI and the FSF. We also examined and studied relevant news articles, blogs, articles, reports, research papers, and books to understand the evolution of FOSS in e-governance in India and various developments within the FOSS movement at the national and international levels. Through this literature review, we laid a foundation for understanding key concepts, fundamental theories, the current state of knowledge and discourse, and historical and emerging trends in FOSS that were relevant to our study.

We also conducted a comprehensive analysis of policies and e-governance initiatives by the GoI that were aimed at promoting FOSS and openness. Through an examination of these policies and the operationalisation of India's e-governance enterprises, we were able to identify their

[8] Policy on Adoption of Open Source Software for Government of India, 2015, https://www.meity.gov.in/writereaddata/files/policy_on_adoption_of_oss.pdf
[9] https://www.un.org/techenvoy/content/digital-public-goods

objectives, approaches, gaps, and limitations, and what that meant for authentically promoting openness in governance.

To gain deeper insights and have a well-rounded perspective from various stakeholders, we conducted semi-structured primary interviews with FOSS contributors and technology consultants at certain government departments at the national and state levels. Through these interviews, we were able to capture the challenges in implementing these policies, managing the digital initiatives of the government, capacity and knowledge gaps within government departments, and other emerging complications in how the government is adopting FOSS principles in its work.

3. Why FOSS

The benefits of using FOSS have been acknowledged and documented by governments worldwide.[10,11]

> "By using open software and working in the open, you remove barriers to critical evaluation and participation from others. Inviting critique from others can be uncomfortable, but it increases the likelihood that the final product is more effective at meeting the needs of multiple stakeholders inside and outside government."[12]

The various advantages of government agencies adopting FOSS are listed below:

3.1. Vendor independence and flexibility

The utilisation of FOSS significantly enhances an organisation's adaptability by enabling the use of multiple vendors and facilitating software sharing among developers and teams. This allows for the use and reuse of software as required. By employing FOSS, full access to the source code is guaranteed to everyone, including the public, without needing any permissions from the vendor. Using FOSS does mean that the onus of service and maintenance lies with other contributors, while government departments still hold the responsibility of ensuring security and

[10] https://gds.blog.gov.uk/2017/09/04/the-benefits-of-coding-in-the-open/
[11] https://digital.canada.ca/2020/02/24/why-open-source-matters/
[12] https://18f.gsa.gov/2018/05/24/what-agencies-have-to-say-about-working-in-the-open/

meeting citizens' needs. However, FOSS makes it easier to attain these goals through a community of contributors who provide mutual support to achieve them. Additionally, FOSS offers the flexibility to switch between various tools that proprietary services rarely offer.

3.2. Greater control of the software

FOSS enables governments to retain control over their software, tools, infrastructure, and services. Many essential software systems used by governments are typically closed-source and proprietary, meaning they are owned entirely by the developer or vendor.

In such cases, only the developer can fix bugs or other issues with the software. Consequently, only the developers possess the ability to address bugs or other software-related issues, such as upgrading, integrating, or making changes in the software. With FOSS, government departments and project owners can easily fix or modify the software. With access to the source code of the software, they can examine the inner workings of the software, modify it, and adapt it to their specific needs. They can also customise the software, fix bugs, and optimise its performance.

3.3. Cost saving and efficiency

Traditional software licences often come with significant costs. FOSS eliminates the need for expensive licensing fees associated with proprietary software. Given the constraint on public funding, government departments are always trying to maximise cost-effectiveness by optimising the use of existing resources. By building on and reusing the work of other departments, organisations, and contributors, the government can decrease the opportunity cost of starting from scratch. Often at the beginning of a project, significant efforts are spent on discovery, understanding users and their needs, and development and testing of various features such as security and usability. By using FOSS, government departments can reduce the time and costs involved in this process and get started on identifying and implementing necessary customisations and modifications. This is a far more efficient approach that also enhances the speed at which software is developed and deployed, thereby reducing overall development costs.

3.4. Greater collaboration and innovation

FOSS provides an open platform for developers world over to collaborate and contribute to the development of software projects. This open collaboration model fosters a diverse and inclusive

community of like-minded individuals where ideas, expertise, and resources can be shared. By working together, discussing, and collectively improving the software, technical teams in governments, alongside developers in the FOSS community, are able to innovate through collective intelligence. By making the source code open and allowing developers to study, modify, and build upon existing solutions, the FOSS approach encourages knowledge sharing and peer learning.

FOSS allows for forking, where developers can create their own versions of a software project and explore various approaches and ideas. By experimenting, contributors are able to take a project in a new direction, introducing innovative features and addressing specific needs that perhaps the original author of the source code may not have thought of. This culture of experimentation embedded in FOSS communities is what enables new ideas to flourish, leading to diversity in solutions.

Enabling collaboration on FOSS projects is much simpler since the initial work of making it open and accessible is already over. Furthermore, the software being open indicates that no special approval, agreement, or tools are required to share or access the code. In contrast, closed source software would require lengthy approval processes for different teams within the same government agency to share work, which could take weeks or even months. With FOSS, collaboration can begin immediately, which proved particularly advantageous in crisis situations such as the COVID-19 pandemic,[13] when governments and the public were able to leverage existing open source software to quickly develop solutions.[14]

3.5. Good software development practices

Open sourcing one's work promotes good development practices through public accountability. It emphasises considerations such as security, software design, documentation, and processes, bringing them to the forefront at the onset of the project. FOSS is often well-documented and supported by a community, which makes it easier for new collaborators to start studying, using,

[13] https://opensource.com/article/20/3/open-source-software-covid19
[14] Sohini Das, "CoWIN Goes Global: India Makes Tech Open Source, 142 Nations Show Interest", *Business Standard*, 6 July 2021, https://www.business-standard.com/article/current-affairs/cowin-goes-global-india-makes-tech-open-source-142-nations-show-interest-121070501046_1.html

and contributing to it. Since all project information is openly available, new team members are able to quickly find what they need. Whether working in-house or outsourced, developers have the freedom to use their preferred tools instead of being limited to proprietary options. Additionally, FOSS collaboration serves as a valuable learning experience for authors and contributors, fostering an exchange of ideas and alternative approaches to solving specific problems. Contributors have the freedom to apply it in ways the original author may not have envisioned, resulting in valuable discoveries for everyone involved.

By embracing open source practices, organisations create opportunities for expansive and creative development built on existing work. Collaborators can utilise open source work as a foundation for addressing similar problems, thereby saving valuable time and preventing redundant efforts. This way, FOSS can enable a partnership between various government agencies and the community to collectively tackle common issues with innovative and imaginative approaches.

3.6. Improved transparency and public accountability

Open access to the software's source code enhances transparency by allowing anyone to examine the code and verify the software's integrity and security. FOSS encourages peer review of the source code by a wide base of users and developers. The community collaboratively reviews, critiques, and contributes to the software's improvement. The collective security and public audit mechanisms enhance the quality, reliability, and functionality of the software. FOSS projects usually have publicly accessible bug reporting and issue tracking where users can report issues they encounter. This open process ensures that flaws and vulnerabilities are acknowledged, tracked, and addressed promptly.

Active discussions, mailing lists, and other modes of open communication are characteristics of active FOSS projects, where contributors can openly discuss and collaborate on the development of the software. Many FOSS projects also have established governance models that actively involve the community in their decision-making processes. These discussions could be about project direction, accepting contributions, and making other decisions transparently. This

transparent and inclusive environment ensures public participation, input, feedback, and collective ownership of the software, subsequently promoting trust and accountability.

4. FOSS policy initiatives by the Government of India

Although the Digital India programme does not explicitly promote the adoption of FOSS, several policies were launched as part of Digital India to leverage FOSS for the development of e-governance applications.

4.1. National Policy on Information Technology, 2012

Released by the Ministry of Electronics and Information Technology (MeitY) in September 2012, the National Policy on Information Technology[15] was aimed at leveraging India's information technology (IT) and information technology enabled services (ITES) sectors to drive India's economic growth. It highlights the role of information and communications technology (ICT) in addressing economic and development challenges across different sectors and situating India as a global hub for IT.

Towards achieving these goals, the policy lists the adoption of open standards and the promotion of open source and open technologies as one of its primary objectives. This makes it one of the first policy documents to mention FOSS as a policy priority for India and lists it as a strategic step for enabling e-governance.

4.1. Policy on Adoption of Open Source Software for India, 2015

In 2015, MeitY released the Policy on Adoption of Open Source Software for the GoI[16] as part of the Digital India programme. Realising that setting up the hardware and software infrastructure to deliver e-governance services would require enormous resources, in an attempt to reduce costs, the GoI wanted to adopt OSS solutions. Therefore, it drew up this policy to institutionalise the promotion and adoption of FOSS technologies as a core component of building e-governance

[15] https://www.meity.gov.in/writereaddata/files/National_20IT_20Policyt%20_20.pdf
[16] https://egovstandards.gov.in/sites/default/files/2021-07/Policy%20on%20Adoption%20of%20Open%20Source%20Software%20for%20Government%20of%20India.pdf

infrastructure. Furthermore, this was a step further in realising the objective mentioned in the National Policy on IT, 2012.

The policy states that all government organisations, at the national and state levels, shall prefer to adopt FOSS solutions over their closed source software (CSS) counterparts when building and implementing e-governance systems. For new e-governance systems and applications and newer versions of legacy systems, government departments are required to specifically include a requirement in their request for proposals (RFP) that suppliers must consider FOSS technologies in addition to CSS solutions when responding. In the event that their responses do not include FOSS alternatives, vendors are required to provide a justification for the exclusion.

Though all government agencies are mandated to comply with the policy, the policy also mentions conditions for exceptions, such as when FOSS solutions fail to meet basic functional requirements for specialised areas of work, when there is a capacity deficit within departments in working with FOSS technologies, or when there is a strategic need to deploy CSS solutions. However, in each of these cases, the government agency is required to provide sufficient justification for the exception.

In addition to these provisions, the policy also introduces the government's role in engaging with the wider FOSS community and including academia and industry through partnerships. It also states that the GoI shall 'actively collaborate'[17] and contribute to FOSS projects at national and international levels.

4.2. Framework for Adoption of Open Source Software in e-Governance Systems, 2015

Published in April 2015, the Framework for Adoption of Open Source Software in e-Governance Systems is a supporting document to the Policy on Adoption of Open Source Software for India, 2015, and provides guidelines for government departments and agencies on how they may select and implement FOSS technologies. The framework is a slightly more technical document compared to the policy where it finds its basis. It serves as a manual for government departments

[17] Policy on Adoption of Open Source Software for Government of India, 2015, https://www.meity.gov.in/writereaddata/files/policy_on_adoption_of_oss.pdf

and agencies to refer to when they are developing and implementing new e-governance systems or upgrading existing ones.

Some of the key areas where the framework provides detailed guidelines are:

a. Areas that are most suited for FOSS adoption.
b. Detailed strengths, weaknesses, opportunities, and threats (SWOT) analysis of adoption FOSS and ways to mitigate weaknesses and threats.
c. Various factors at economic, security, technological, organisational, environmental, and individual levels can positively and negatively influence the uptake of FOSS.
d. Different types of support models that are available when implementing FOSS and the advantages and disadvantages associated with each of these models.
e. Available FOSS licences and guidelines for selecting the appropriate type of OSS libraries based on the licence type.
f. Security aspects to consider when deploying FOSS.
g. Guidelines on procurement, including factors to consider and include in RFPs in addition to the required specification for preference for FOSS.
h. Business metrics that can be used to rate FOSS against CSS.
i. The various stages through which FOSS solutions must go through when integrating into new or existing e-governance systems.
j. Approaches to create an ecosystem level change to promote FOSS through creation of institutional mechanisms and partnerships with industry, academia, and FOSS communities at national and international level

Additionally, the framework document also provides examples of FOSS stacks for various use cases and an illustrative list of FOSS that can be adopted by government agencies. Going beyond guidelines and instructions, the framework also discusses various issues that are important to consider when thinking of FOSS in e-governance applications, such as unified software development for commonly used devices, use of FOSS-friendly web browsers, localisation, and a rapid application development environment that is conducive to FOSS.

4.3. Policy on Open Standards for e-Governance, 2010

Recognising the importance of standards in ensuring that there is a seamless exchange of information across various government systems, MeitY notified the Policy on Open Standards for e-Governance[18] in November 2010. The primary objective of the policy was to facilitate interoperability among different hardware and software used by government agencies. In order to meet this objective, the policy directed GoI to adopt 'Single and Royalty-Free (RF)' open standards.

The policy is applicable to all systems at the interface and data-archival levels. The responsibility of adhering to this policy lies on application owners, who must ensure that legacy systems and their newer versions must use open standards for interoperability.

It lays down the criteria to qualify as an 'open standard'. As per the document, for an open standard, its specification document should be available with or without a nominal fee, the patent to implement it should be available on a royalty-free basis, and its stewardship should lie with a non-profit organisation. It should also be open, technology-neutral, and should support localisation. The policy goes on to further list the criteria that can be used to decide which open standard to adopt when presented with multiple options.

The policy also has guidelines for choosing a standard when it fails to meet the bare minimum criteria to be an open standard. In such a scenario, the policy recommends the adoption of an interim standard that can be temporarily adopted after relaxing the mandatory criteria. Emphasising that preference shall be given to functional and technical requirements of the GoI, provisions in the policy offer room for flexibility and exceptions in selecting standards based on availability and requirements for specific purposes. It also allows the creation of standards by a designated body for cases where there are no standards available. This designated body appointed by the GoI is responsible for reviewing and recommending standards and initiating the formulation of interim standards.

---

18 https://egovstandards.gov.in/sites/default/files/2021-07/Policy%20on%20Open%20Standards%20for%20e-Governance.pdf

4.4. Manual on the Implementation of Policy on Open Standards for e-Governance, 2010

An extension of the Policy on Open Standards for e-Governance, the Manual on the Procedure for Implementation of Policy on Open Standards for e-Governance,[19] published by MeitY in November 2010, provides guidelines to implement the policy, select open standards, and provides more context on the rationale for having the policy in the first place.

Going beyond the list of criteria mentioned in the policy, the manual provides details on mechanisms to implement the policy, focusing on the processes and roles of various stakeholders like the expert committee and designated body in executing the policy. As per the policy, GoI is also required to openly publish an updated list of open standards and recommendations on which standards to adopt. Furthermore, government agencies are required to specify requirements in their RFPs for e-governance projects that mandate the adoption of open standards. As part of the policy implementation, GoI is to manage all activities related to compliance, comparison, and testing of competing standards when implementing e-governance projects.

4.5. Interoperability Framework for e-Governance, 2015

The Interoperability Framework for e-Governance (IFEG)[20] was released by MeitY in October 2015 under the Digital India initiative as part of the National e-Governance Plan (NeGP). The IFEG was aimed at furthering the NeGP vision of integrating e-governance initiatives across India and establishing a more holistic approach to the delivery of public services. It is worthwhile to note that DeitY had notified the Technical Standards for Interoperability Framework for e-Governance in India[21] in May 2012, which provided a detailed list of technical standards for various domains.

The IFEG proposes different ways to establish interoperability and information exchange between e-governance systems, along with recommended steps that can be undertaken by public agencies. Unlike previously mentioned policies, IFEG is more of an advisory guideline where

---

19 https://egovstandards.gov.in/sites/default/files/2021-07/Manual%20on%20the%20Policy%20on%20Open%20Standards.pdf

20 https://egovstandards.gov.in/sites/default/files/2021-07/Interoperability%20Framework%20For%20e-Governance%20%28IFEG%29%20Ver.1.0.pdf

21 https://egovstandards.gov.in/sites/default/files/2021-07/Technical%20Standards%20for%20IFEG%20Ver1.0.pdf

government agencies can choose to implement the framework when implementing or upgrading their e-governance systems. However, at odds with the optional nature of compliance, the framework does list out consequences for non-compliance, such as refusal to establish connections with government infrastructure such as the state-wide area network (SWAN), state data centre (SDC), and national data centre (NDC) and restricted use of core components in systems.

There are three different levels of interoperability addressed in the framework – organisational, semantic, and technical. Acknowledging that these different levels of interoperability are influenced by political, legal, socio-cultural, and other factors, the framework lays emphasis on the need for interoperability mechanisms to be designed in a participatory way and executed in a 'transparent, consensual, and collaborative' manner. It goes into further detail to explain the level of interoperability at each level, the steps required to achieve them, and the challenges that one might encounter when trying to implement them.

4.7. Policy on Open Application Programming Interfaces (APIs) for Government of India, 2015

Recognising the significance of application programming interfaces (APIs) in promoting interoperability, the GoI notified the Policy on Open APIs for government organisations[22] in June 2015. Through this policy, the aim was to formally institutionalise the use of open APIs in government organisations by mandating them to do so when implementing various e-governance software applications. It is applicable to all government organisations under the jurisdictions of the union and state governments and to all e-governance systems and applications.

According to the policy, an open API should satisfy the following conditions:

- Information shared by government agencies, including data and functionalities of their e-governance applications, should be open and machine-readable.

[22] https://www.meity.gov.in/writereaddata/files/Policy%20for%20API%20for%20GoI.pdf

- In line with the National Data Sharing and Accessibility Policy, 2012,[23] all relevant information published by government organisations should be publicly accessible and free of charge when possible.
- Comply with the National Cyber Security Policy and other security policies and guidelines.
- Shall be stable, scalable, and platform and language agnostic

- Any government agency that is a consumer of APIs provided by another government organisation shall adhere to the rules for handling, authenticating, and authorising data and information as prescribed by the API publishing organisation.
- For developers to be able to use the API, it should be provided with sufficient documentation, including sample code.
- It should be backwards compatible with at least two of its earlier versions, and its life cycle shall be shared openly.

In addition to publishing open APIs to enable integration with e-governance applications and systems, government departments are required to integrate their systems and applications with each other and use open standards identified and notified by the GoI. The policy also states the establishment of an implementation committee to implement provisions of the policy and provide support mechanisms to help manage the APIs of various organisations. Similar to the previously mentioned policies, under the open API policy, RFPs by government departments must include a requirement to publish APIs that shall be available to all.

4.8. Policy on Collaborative Application Development by Opening the Source Code of Government Applications, 2015

The push for Digital India resulted in several government departments at the national and state levels developing e-governance initiatives. With many of these digital initiatives trying to address similar problems, the GoI realised that instead of starting from scratch and reinventing the wheel each time, government departments could benefit from sharing code and building e-governance software and applications in a collaborative manner. Therefore, in February 2015,

[23] "National Data Sharing and Accessibility Policy, 2012", *The Gazette of India*, 17–12 March 2012, pp 93–99, https://dst.gov.in/sites/default/files/gazetteNotificationNDSAP.pdf

MeitY notified the Policy on Collaborative Application Development by Opening the Source Code of Government Applications.[24] In addition to optimising resources and improving efficiency, through this policy, the government also hoped to standardise e-governance products.

According to the policy, the government has intellectual property rights (IPR) to the source code of the software that was custom-built or customised for the government agency, developed by the agency itself or by a private organisation. For commercial-off-the-shelf (COTS) products, the government would retain full rights to any customised source code, including any add-on modules and plug-ins. In addition to access to the source code, a government organisation owning the IPR would imply that they have the right to reuse the customised source code and share it with another government department or public-sector institution. As per the policy, these rights shall be granted to the government by specifying this as a requirement during the procurement process for custom-built applications and through clauses in the contract between the government and the private agency for COTS products. For e-governance applications with a source code already openly available, the government would own unlimited rights to any modifications of the source code.

To instil an environment of collaboration, the policy envisions a collaborative application development platform where government departments can host and share the source code to their e-governance applications. The platform would have features such as version control and code validation. To encourage the adoption of this platform, the policy also mentions that when developing or procuring new software, existing software applications available on the platform shall be given preference. While not mandatory for legacy systems, the policy recommends sharing the source code and object code of the application along with the database schema, installation document, installation script, and other documents that might be useful. The policy also hints at a set of guidelines covering aspects such as security, governance frameworks, and licensing policy for government agencies to follow when using the platform. However, no such guidelines were notified or drafted.

[24] https://www.meity.gov.in/sites/upload_files/dit/files/policy_government_application.pdf

This policy is applicable to all e-governance software applications, products, and components, regardless of whether they are being developed in-house or procured. However, it does not limit government departments from pursuing commercial interests. Moreover, e-governance and applications that have consequences on national security or have strategic importance are exempt from this policy.

While there are several policy documents that bring attention to the use of digital technologies for governance, the mentioned policy documents explicitly highlight the significance and benefits of adopting FOSS technologies in e-governance. These policies collectively form the core policy environment to institutionalise the adoption of FOSS at the union government level. They lay the groundwork for educating implementors of e-governance initiatives about the importance and relevance of using FOSS technologies for developing software and applications while also providing guidance, illustrative examples, and case studies of how government entities can integrate the principles set out by these policies into their daily workflow and operations.

5. Contemporary policies on openness in e-governance systems and applications

More recent policy documents that encourage FOSS adoption for e-governance include the Strategy for National Open Digital Ecosystems (NODE) and the India Enterprise Architecture (IndEA) framework. While these are not fully formed policies themselves, these documents propose frameworks, strategies, and guidelines to not just adopt open source technologies and standards but for these e-governance products and components to be open themselves. While the earlier policies emphasise the use of FOSS for interoperability – and hence, improved efficiency and collaboration among government departments – these newer policy documents advocate for standardisation and interoperability as a core tenet. Each of these policies envisions digital public services offered and used by government departments in a whole-of-government model,[25] as emphasised in the United Nations e-Government Survey, 2016.[26]

[25] "Whole-of-government", "one-stop government", or "joined-up government" is the approach where a government actively uses formal and informal networks across different agencies and departments within the government to eliminate isolated silos in public administration and increase integration, coordination, and capacity.
[26] https://publicadministration.un.org/egovkb/en-us/reports/un-e-government-survey-2016

5.1. India Enterprise Architecture Framework (IndEA)

Recognising the fragmentation across various sectors, departments, and levels of government, the IndEA Framework[27] introduced by MeitY in October 2018 provides a framework that is capable of attaining the much-needed interoperability and integration in e-governance. The vision of the framework is "to establish best-in-class architectural governance, processes, and practices with optimal utilization of ICT infrastructure and applications to offer ONE Government experience to the citizens and businesses." The "one' government" model intends to provide a service delivery mechanism where citizens have access to government services through a single interface by streamlining and standardising the processes and systems employed in e-governance efforts. It introduces the concept of 'virtualisation of departments' by proposing a boundaryless flow of information.

The Working Group on National Enterprise Architecture by MeitY, constituted by GoI, designed the IndEA framework. The title 'IndEA' is a misnomer because what IndEA proposes is a framework for building an enterprise architecture but not a blueprint to build the architecture itself. The framework sets forth eight reference models (RMs) for each specific domain within the enterprise architecture. These RMs are abstract collections of best practices that can be used to make technology and design choices by e-governance project teams. Typically, an RM recommends a list of standards to comply with for a particular domain, a relationship with other RMs, and communication across these different RMs throughout the enterprise architecture. The various domains that these RMs are listed for are performance, business, application, data, technology, integration, security, and enterprise architecture governance. Every RM has three common characteristics with other RMs, namely, its abstract nature, compliance with domain-specific standards, and neutrality towards technology.

The framework mentions 'openness and transparency' as a core principle towards the integration of technologies and recommends the use of open standards, open source technologies, and open APIs. This principle is particularly reflected in the application reference model, technology reference model, and integration reference model.

[27] https://www.meity.gov.in/writereaddata/files/IndEA_Framework_1.0.pdf

a. Application reference model (ARM): The primary objective of ARM is to classify similar applications and make them available for maximum reuse. To achieve this, ARM suggests the use of open APIs and open source software as part of its guidelines and best practices. According to ARM, for software applications to be compatible with the application domain within the enterprise architecture, they must conform to open standards. For original equipment manufacturer (OEM) products procured by government entities, ARM states that it must be accompanied by a vendor-neutral API. Furthermore, government agencies must ensure that at least two independent OEM products are available with the same vendor-neutral API or standard.
b. Technology reference model (TRM): The TRM provides an end-to-end layout for technology architecture and lists all the components that would be part of such a technology system. For each of these technology components, a TRM is required to identify and delineate all applicable and relevant open standards, open formats, and open source products to be included in the technology architecture. One of the guiding principles of TRM specifically states, "Open Standards are adopted in the design and implementation of all greenfield systems. Legacy systems are incentivized to migrate to open standards, where required." Referring to the 2015 Policy on Adoption of Open Source Software for India, TRM reiterates that preference should be given to open source products where applicable. It also emphasises the need for open APIs as part of the TRM schematic in order to effectively create an environment for integrating with third-party applications.
c. Integration reference model (IRM): The IRM layer is described as the 'glue' that enables coordination across various government institutions and agencies, eventually providing a seamless experience to citizens when accessing e-governance services. Given the diversity and complexity across various levels of government, both horizontally and vertically, the IRM seeks to provide functional as well as political and legal integration across technology infrastructure, applications, data, business processes, and services that collectively form the backend operations of e-governance efforts. A key guiding principle of the IRM to achieve this byzantine goal, as listed in the document, is 'openness and transparency'.

The framework is also accompanied by an IndEA adoption guide,[28] which was published by MeitY in October 2018. Developed by the Standardisation Testing and Quality Certification Directorate (STQC)[29] within MeitY, with contributions from the National Informatics Centre (NIC),[30] MeitY, and the Centre for Development of Advanced Computing (CDAC),[31] the guide's objective is to support government entities in integrating the IndEA framework and RMs at various stages of an e-governance project. The guide lists the roles and responsibilities of various entities in the implementation of the IndEA framework. With NIC as the nodal agency, the guide proposes establishing an IndEA Centre for Excellence to assist state governments and other government agencies at the national and sub-national levels that are looking to develop and implement enterprise architectures. Acknowledging the role of capacity building as critical in the success of implementing the framework and the guide, it suggests upskilling the State e-Mission Teams (SeMTs)[32] and promoting industry–academia collaboration towards developing pedagogy for capacity-building and research.

Following the paradigm shift in the e-governance domain from systems to ecosystems, as proposed in the Strategy for National Open Digital Ecosystems (NODEs), MeitY updated the 2018 IndEA framework and published the IndEA 2.0 framework.[33] The 2.0 version of the IndEA framework, currently in draft form, was prepared by a steering committee constituted by MeitY. Building on the 2018 IndEA framework and the NODE approach, IndEA 2.0 proposes the development of enterprise architecture using a set of building blocks and leveraging ecosystem players. The building block is a technology component that represents a business or technical functionality and is intended to be stable, reusable, and scalable. It also lists a set of 27 principles across five categories, namely, ecosystem, architecture, business, technology, and architecture governance. Some of these principles that champion the principles of FOSS are:

a. Open API-based: Building blocks must be 'open API by default' and must comply with the 2015 Policy on Open APIs for the Government of India.

[28] https://www.meity.gov.in/writereaddata/files/IndEA_Adoption_Guide10-Oct2018.pdf
[29] https://www.stqc.gov.in/
[30] https://www.nic.in/
[31] https://cdac.in/
[32] https://negd.gov.in/state-e-mission-teams/
[33] https://d32jquim0n1d64.cloudfront.net/2023/04/InDEA-2_0-Report-Draft-V6-24-Jan-22.pdf

b. Open, open, and open: All digital systems designed under the IndEA framework should be built on open source, published as open source, and must conform to open standards. E-governance products, whether developed or procured, must comply with the 2015 Policy on Open Source Software for the Government of India. In line with this policy, any exceptions to this principle shall have to be justified.
c. Participatory design: To encourage participatory design of e-governance initiatives, open communities shall be promoted.

The IndEA 2.0 framework also proposes to embody an open digital architecture comprising three core layers, namely, identities, assets, and transactions. Such an architecture utilises verifiable credentials, open networks, and open protocols. For the development and implementation of these layers, the framework recommends using open source implementations to issue credentials and opening the development and staging[34] of environments to promote the collaborative development of technologies and applications. An open network is one that facilitates interoperability and interconnection across several applications and platforms using open protocol specifications. The framework also highlights the need for government agencies to interact with relevant open source communities in implementing open protocols and driving open networks.

Notable examples of open digital architecture are discussed in the next section.

5.2. Strategy for National Open Digital Ecosystems (NODEs)

The National Open Data Ecosystem (NODE) strategy whitepaper, released by MeitY in March 2020[35], classifies the evolution of using information and digital technologies for governance as GovTech 1.0, GovTech 2.0, and GovTech 3.0. It labels the use of IndiaStack and IndEA framework in the architecture of e-governance applications as GovTech 3.0. This GovTech 3.2 era focuses on having an 'ecosystem-based approach' to building digital governance platforms NODEs are defined as "open and secure delivery platforms, anchored by transparent governance mechanisms, which enable a community of partners to unlock innovative solutions, to transform societal outcomes." The document emphasises the significance of having these platforms 'open'

[34] A staging environment (stage) is a nearly exact replica of a production environment for software testing.
[35] https://www.medianama.com/wp-content/uploads/mygov_1582193114515532211.pdf

to allow for multi-stakeholder collaboration and inter-departmental integration, which is also promoted as a unique feature of the NODE approach.

The dilution of FOSS principles in these inward-facing policies[36] is also characterised by the positioning of interoperability as a substitute for openness. Even in these contemporary policies, the larger objective has been towards seamless integrations of data and technologies across government departments at both the central and state levels.

6. FOSS adoption in GovTech

Since the advent of Digital India, the GoI, through MeitY, has made several investments in developing the FOSS Infrastructure in India. Noteworthy initiatives include setting up the National Resource Centre for FOSS (NRCFOSS) and Bharat Operating System Solutions (BOSS), both of which were behind the successful implementation and deployment of the following major FOSS initiatives:

6.1. Bharat Operating System Solutions (BOSS)

Indigenous GNU/Linux-based operating systems whose distribution is certified by the Linux Foundation.

6.2. EduBOSS

The educational variant of BOSS is a full-featured, user-friendly Linux operating system that comes with pre-installed educational applications, games, paint and graphic tools, typing tutor, and a host of tools and packages for learning and teaching that are useful for schools.

6.3. BOSS server

The BOSS server is a light-weight version of Debian-based GNU/Linux with necessary security features enabled and designed for hosting web servers, proxy servers, mail servers, network servers, database servers, file and print servers, virtualisation servers, etc. It has been particularly useful for subject-matter experts and government organisations for hosting their websites and internal servers.

[36] Apar Gupta, "Analysis of FOSS Government Policies in India", 25 June 2022, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4146240

6.4. Meghdoot

A cloud suite developed by the GoI using free and open source software to offer cloud services.

6.5. Swar-Suchak

A voice-enabled information retrieval system in multiple languages. The voice gateway successfully integrates the mobile telephone network with an automatic speech recognition system, a text-to-speech system for Hindi and English, and a web navigation system based on open standards using open source software alone.

6.6. GEstures with Mouse (GEM)

An input mechanism developed for Linux desktops that uses gestures to give input to the system. Gestures can be drawn using a mouse or an equivalent device such as a touchpad or joystick. GEM provides support for continuous as well as discontinuous gestures.

6.7. Anumaan

Anumaan is an open source, on-screen predictive text entry system designed to aid people with motor disabilities. It has been integrated with Linux desktops and is also available as a stand-alone tool. Anumaan provides an easy-to-use interface for incorporating predictions in text.

6.8. ALViC

Accessible Linux for Visually Challenged (ALViC) is a software-based assistive technology and solution developed for differently-abled people. The main aim is to make computer usage easier and affordable to empower differently-abled people to access ICT services efficiently and effectively.

6.9. Creative Computing @ Schools

An educational e-journal for the Central Board of Secondary Education to share teachers' educational pedagogical experiences and students' innovative ideas. As an e-product, it is cost-effective and has the advantage of quick and definite delivery to the student and teacher community.

6.10. Integrated Library Management System (ILMS)

The ILMS is the most advanced and cost-effective open source automation solution for public, academic, and digital video libraries with comprehensive functionality, successfully integrating Koha and DSpace.

Despite the success of these initiatives suggesting the government's commitment to FOSS, they remained isolated and did not create a wide-scale impact towards greater FOSS adoption.

7. OpenForge

Pursuant to the Policy on Open Application Programming Interfaces (APIs) for Government of India, 2015,[37] in March 2017, the National e-Governance Division (NeGD) of MeitY launched OpenForge, a platform to host FOSS and serve as an Indian alternative to Github and SourceForge, with the aim of promoting openness and collaboration in the development of software applications for e-governance. OpenForge was developed in light of the Digital India push, where government departments across the union and state governments were mandated to develop software applications for the purposes of e-governance. By having a single go-to platform for e-governance code repositories, the objective of OpenForge was to encourage code sharing of e-governance applications and, thereby, promote the use of open source software. The primary aim of building OpenForge was to break the culture of silos that persists in government operations and improve efficiency while reducing costs and time incurred in developing applications from scratch.[38]

OpenForge is based on Tuleap,[39] an open source tool that allows end-to-end tracking and management of an application lifecycle. OpenForge also uses an open source scripting language to accept and maintain various projects on the platform. It has two code-sharing models – Government to Government (G2G) and Government to Community (G2C).[40] In the G2G model, sharing of the e-governance–related source code is restricted to government departments and subject to approval by project managers and administrators, while in the G2C model, community

[37] https://www.meity.gov.in/writereaddata/files/Policy%20for%20API%20for%20GoI.pdf
[38] https://openforge.gov.in/openforge/about.php
[39] https://www.tuleap.org/
[40] https://openforge.gov.in/openforge/about.php

members can participate in and contribute to projects, subject to approval by project administrators.

In addition to restrictions on 'suitability' for participation, the platform does not allow projects that are unrelated to e-governance and allows government entities to host private repositories. Hence, the decision on whether or not to open up the source code for community examination or to even use OpenForge remains at the discretion of the project owner or relevant government department. Currently, projects such as DigiLocker,[41] Government e-Marketplace,[42] and Unified Mobile Application for New Age Governance[43] are hosted on the OpenForge platform as private repositories.

With these conditions and restrictions, it is evident that OpenForge is not the paradigm shift in creating a culture of openness that it was claimed to be. As "India's GitHub for e-Governance projects,"[44] OpenForge was projected as a technology infrastructure aimed at fostering the adoption of FOSS among various government agencies. The impact of OpenForge in improving efficiency in software development, code sharing and reuse across various government projects, or cost reduction is yet to be measured.

8. IndiaStack

Unlike the earlier FOSS initiatives part of GovTech 2.0 that had a sound policy basis for their development, the policy foundations of GovTech 3.0 pose a chicken–egg problem – it is unclear which came first, the policy or the blueprint for technological architecture. Significant e-governance initiatives to emerge from the GovTech 3.0 era have their technical foundations in the IndiaStack architecture, which have been defined as a "set of open APIs and digital public goods" that are aimed at solving population-scale problems in the areas of "identity, data and payments".[45] As per the definition of digital public goods (DPGs), it is understood that

[41] https://www.digilocker.gov.in/
[42] https://gem.gov.in/
[43] https://web.umang.gov.in/landing/
[44] Jagmeet Singh, "OpenForge Debuts ss India's Github for E-Governance Projects", opensourceforu.com, 7 March 2017, https://www.opensourceforu.com/2017/03/openforge-debuts-indias-github-e-governance-projects/
[45] https://indiastack.org/

components of IndiaStack shall be “open source software, open data, open AI models, open standards, and open content”.[46]

Over the years, there have been several mentions of the success of digital ecosystems built using IndiaStack, of which, the following are the most prominent:

8.1. Aadhaar

The 12-digit unique identity system is built using the Modular Open Source Identity Platform (MOSIP)[47] and licensed under Mozilla Public License 2.0. The MOSIP is a digital identity system that allows governments, organisations, and developers to create identification solutions. It is anchored by the International Institute of Information Technology, Bangalore (IIIT-B) and funded by philanthropic organisations such as the Bill and Melinda Gates Foundation, Omidyar Network, and Tata Trusts.

The documentation and codebase for MOSIP are available on GitHub[48] for anyone to view and contribute. There are additional materials and videos provided on YouTube and Discord to offer support and assist with onboarding. Currently, there is no Software Bill of Materials (SBOM)[49] for MOSIP. Assessing the extent of openness of MOSIP would require extensive technical scrutiny of the code repositories.

It is worthwhile to note that despite the scale and extent of the impact of the MOSIP project, the repository with the highest number of Github stars[50] is the documentation repository – with 29 stars and 100 forks. The reasons for limited community engagement are unknown and require further investigation to identify barriers and apprehensions that community members might be facing.

---

[46] https://digitalpublicgoods.net/DPGA_Strategy_2021-2026.pdf
[47] https://mosip.io/index.php
[48] https://github.com/mosip
[49] Software Bill of Materials (SBOM) is a list of all the open source and third-party components present in a codebase. Typically, handy for identifying security risks, an SBOM lists the licenses that govern the various components, the versions of the components used in the codebase, and their patch status.
[50] Github stars are a proxy for community contributions to a particular repository.

8.2. Unified Payments Interface (UPI)

Developed by the National Payments Corporation of India (NPCI),[51] UPI is a real-time payment system that facilitates instant peer-to-peer (P2P) and peer-to-merchant (P2M) transactions. UPI is a technology architecture framework with a set of standard API specifications that provides a single interface across all NPCI systems, thereby enabling interoperability and facilitating online payments.

8.3. Co-WIN

The Co-WIN vaccination platform developed in light of the COVID pandemic was developed by the Ministry of Health and Family Welfare (MoHFW) and the National e-Governance Division of MeitY. The platform is claimed to have been developed using open source solutions, but it is not an open source platform since its source code is available to authorised third parties alone.[52] An SBOM or list of open source technologies used to develop the Co-WIN platform is also unknown. However, its APIs were made open and are publicly accessible. DigiLocker DigiLocker is a cloud storage solution for digital documents. Built using open source tools like ownCloud server, PHP, Python, and Node.js, DigiLocker is supposedly the largest government project based on open source technologies.[53] On the server front, DigiLocker uses Nginx and Apache; for gateway access, MonoDB is used; for user account-related metadata, MariaDB is deployed; and for the gateway, the NoSQL database is used.[54] Open source frameworks like Nginx and Memcached have been used for high scalability. The DigiLocker technical specifications document also states that the entire system has been kept "open via common standards".[55]

---

[51] https://www.npci.org.in/
[52] Swathi Moorthy, "Is Co-WIN an Open Source Platform?", moneycontrol.com, 5 October 2021, https://www.moneycontrol.com/news/business/is-co-win-an-open-source-platform-7542531.html
[53] Jagmeet Singh and Aashima Sharma, "Government Leverages Open Source to Build DigiLocker for Indian Citizens", opensourceforu.com, 4 October 2016, https://www.opensourceforu.com/2016/10/government-leverages-open-source-build-digilocker-indian-citizens/
[54] Jagmeet Singh and Aashima Sharma, "Government Leverages Open Source to Build DigiLocker for Indian Citizens", opensourceforu.com, 4 October 2016, https://www.opensourceforu.com/2016/10/government-leverages-open-source-build-digilocker-indian-citizens/
[55] https://img1.digitallocker.gov.in/assets/img/technical-specifications-dlts-ver-2.3.pdf

8.4. Aarogya Setu

The source code of the COVID-19 contact-tracing app developed by the National Informatics Centre (NIC) of MeitY was released on the OpenForge platform.[56] However, only the client side of the source code was released on OpenForge for public view and not the server side.[57] As per the source code released, some of the libraries and modules that the code depends on are not yet released. Furthermore, there is no documentation available on how files are executed on a server or on the technology's architecture.

8.5. Digital Infrastructure for Knowledge Sharing (DIKSHA)

DIKSHA is built using Sunbird Ed,[58] which is available under the MIT license allows for each state to leverage the platform according to their specific requirements in addition to using a list of Sunbird modular tools. The DIKSHA website provides a list of the various open source technologies used.[59]

8.6. Unified Mobile Application for New Age Governance (UMANG)

UMANG is a mobile application that aims to provide access to a variety of government services and schemes from central, state, and local government departments on a single unified platform for citizens. Built on the IndiaStack framework and considered a DPG, the UMANG platform is based on an open source technology stack.[60] "Designing open standards based, scalable (cloud hosted), modular/loosely-coupled, secure and preferably open-source stack" has been listed as part of its implementation methodology.[61] However, information regarding the open standards and open source technologies deployed as part of UMANG's technology architecture remains undisclosed.

8.7. E-Sanjeevani

E-Sanjeevani is a cloud-based real-time telemedicine platform that was developed by the Centre for Development of Advanced Computing (C-DAC) based on the IndiaStack framework, "using

[56] "Backend Code of Aarogya Setu Released in Open Domain", PIB, 20 November 2020, https://pib.gov.in/PressReleaseIframePage.aspx?PRID=1674492
[57] https://privacy.sflc.in/aarogyasetubackend/
[58] https://github.com/Sunbird-Ed
[59] https://diksha.gov.in/help/diksha-oss/
[60] https://www.hcimbabane.gov.in/docs/16000624571.pdf
[61] https://www.indiastack.global/umang/

open-source tools/stack".[62] While a blueprint of the high-level architecture used to develop the platform is available, information on what open source technologies were used is unavailable.

8.8. Government e-Marketplace (GeM)

GeM is an online platform developed by the GoI to facilitate the procurement of goods and services by government departments, organisations, and public-sector undertakings. It has been "developed as a full stack Open Source application platform" but is itself not open source and does not provide any information regarding the open source tools deployed in the platform's architecture. As opposed to the earlier policy documents that mandate preference for adopting FOSS in building e-governance applications, none of the technical specifications of building the GeM platform listed in the memorandums of understanding with technology partners specifies a requirement to propose FOSS alternatives or justify the lack of FOSS solutions in the proposal.

8.9. e-Courts

The core periphery model of the Case Information Software (CIS) used in phase II of e-Courts uses 'Unified as National' as the core and each High Court as the periphery. The CIS was developed by NIC based on FOSS solutions to ensure software compatibility and interoperability.[63] The National Judicial Data Grid (NJDG) developed as part of the e-Courts project has open APIs that allow government departments to access the NJDG data.

Each of the listed e-governance applications has claimed to use FOSS technologies for its development. However, it is critical to draw the distinction between the adoption of FOSS tools and products versus being a FOSS product itself. With the exception of Aarogya Setu, all the IndiaStack initiatives discussed in this study are closed source themselves. Even in the case of Aarogya Setu, it is selectively open, redacting major information regarding the backend functioning of the platform. While the adoption of FOSS is also commendable, it is also important to disclose to what extent FOSS technologies have been employed within a particular application. It is evident that none of the discussed platforms are open source.

[62] https://www.indiastack.global/esanjeevani/
[63] https://ecourts.gov.in/ecourts_home/static/about-us.php

9. Discussion

9.1. Poor implementation of FOSS policies

While FOSS policies and initiatives reflect a commitment to promoting open source software and its benefits, there are severe limitations and failures during the implementation stages that warrant a critical analysis of the policies themselves.

Despite the guidelines and directives issued by MeitY, the actual adoption and implementation of FOSS policies have been slow and inconsistent across various government departments. The absence of clear mechanisms and incentives to ensure compliance with these policies has resulted in limited progress.

Many policymakers, decision-makers, and stakeholders in India's e-governance sector may not have a comprehensive understanding of the benefits, potentials, and technical aspects of FOSS. While the policy documents delineate and demystify a lot of these concepts, not providing sufficient awareness and training programmes for government officials and IT professionals within e-governance project teams has hampered the adoption and utilisation of FOSS in solution delivery.

9.2.Open washing

Open washing in the open source movement was first identified by Michelle Thorne in 2009.[64] The concept is similar to greenwashing,[65] and is defined as a phenomenon that occurs when a product, process, or data is called 'open' but is not actually open. Broadly speaking, in the open source ecosystem, open washing is claiming to be open source without necessarily providing an environment that allows developers to exercise their freedom to use that software.

In the government, open washing is practised when government departments or ministers publicise software products as open source when they are actually not licensed under an open source license or are built using FOSS products and tools but do not make their source code

[64] https://michellethorne.cc/2009/03/openwashing/
[65] https://en.wikipedia.org/wiki/Greenwash

available. For instance, the OpenForge website states that, "by opening the source code, the Government wants to encourage collaborative development between Government departments / agencies and private organizations, citizens and developers to spur the creation of innovative e-Governance applications and services",[66] when it primarily promotes G2G code sharing that is 'private' and closed in nature. The OpenForge initiative has confused code sharing with opening the source code while failing to adhere to the core principles and criteria of FOSS. The very use of the term 'open' in OpenForge is misleading, more so because it is a closed source software.

Another instance of open washing is the announcement from NPCI on introducing the BHIM open source license model.[67] Under this licensing model, the source code for the Bharat Interface for Money (BHIM) app would be made available to UPI ecosystem players that did not have a UPI app of their own. 'Open source' is a term for software that has been licensed under OSI-approved licenses. The open source definition (OSD) criteria for the distribution of license[68] clearly states that, "The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties." Deliberately creating an in-house open source license despite the existence of plenty of already OSI-approved licenses is reason enough to be suspicious that the act is a subterfuge to open wash.

At the global level, Co-WIN was marketed as an open source application that was available to all countries.[69] Repeatedly, the official website labels Co-WIN as open when, in reality, only the APIs have been made open and the source code is available only to authorised partners. The use of the term 'open' to describe Co-WIN is grossly misleading and yet another case of open washing.

Displaying support for transparency and accountability by claiming to be open source code of government applications while having only a few that are publicly available and licensed as open

[66] https://openforge.gov.in/openforge/about.php
[67] "NPCI Introduces 'BHIM App Open Source License Model'", NCPI, 9 November 2022, https://www.npci.org.in/PDF/npci/press-releases/2022/NPCI-Press-Release-BHIM-App-Open-source-license-model.pdf
[68] https://opensource.org/osd/
[69] "PM Addresses CoWin Global Conclave as India Offers CoWin Platform as a Digital Public Good to the World to Combat COVID19", PIB, 5 July 2021, https://pib.gov.in/PressReleasePage.aspx?PRID=1732812

source software is a misappropriation of the term ‘open’. Whether or not the software is open can be a critical factor in deciding its adoption by a user. Just like false statements in commercial advertising are deceptive in a material way, the GoI’s attempts at open washing and claiming software as ‘free and open source’ are also duplicitous.

9.3. Open yet closed

The term ‘open’ is in itself quite open-ended. The contemporary discourse around the open source movement has highlighted the shortcomings in the traditional definitions and criteria listed by OSI and FSF and states that a license alone cannot establish something as open source. Fundamentally, an open source license grants one the right to fork, i.e., copy and modify code, and should be seen as a first step towards making software truly open source. The most inherent feature and arguably one of the greatest assets of the open source movement that is responsible for unlocking the value of open source software is the community that it fosters. The open source community is what drives collaboration and collective innovation. Today, FOSS is not just about licenses; it is about the community. ‘Open’ in open source now also stands for ‘open for collaboration’.[70] After all, there is no point in FOSS if there is no community built around the software to add value to it.

OpenForge’s failure to drive engagement with the FOSS community could be attributed to the lack of an ecosystem. Indian FOSS community members have pointed out that instead of going out of their way to reinvent the wheel, the GoI could have leveraged the GitHub and government portal[71] to encourage collaboration with the community since GitHub already has a large userbase. Software developers have also shared that OpenForge is not ‘open enough’[72] for them to participate and that the purpose of GitHub as a platform is not just to provide but to provide an ecosystem where community discussions can thrive.

Circumventing the letter of the open source definition (OSD) or free software definition (FSD) is not new. In the past, organisations have played around the openness of a project by meeting the criteria set out by OSD and FSD but not really allowing collaboration. A common theme across

[70] https://blog.opensource.org/open-yet-closed/
[71] https://government.github.com/
[72] Alok Soni, “How Open is OpenForge”, yourstory.com, 12 June 2017, https://yourstory.com/2017/06/openforge

the government's initiatives that are 'open' is that there is a lot of structure and restriction around maintaining control of the project. Currently, FOSS initiatives by the GoI, regardless of whether they were being built under GovTech 2.0 or GovTech 3.0, operate as proprietary software in terms of distribution and cooperation despite being OSI-licensed. By restricting collaboration, one is implicitly undervaluing the benefits of open source.

10. Conclusion

It is a lot easier to tick the 'open' box than to actually work in the open. Open source aspires to be a new way of thinking about ownership and accountability, something built by and responsive to the collective of users rather than purely traditional market mechanisms. Adopting open source practices means moving the culture away from one of proprietary holdings and closed ownership and towards collaboration and working in the open. Working in the open involves performing the actual work – the individual code changes, the code reviews, discussions, project management, and more – in the open for anyone to see. The benefits of this process, such as improving transparency, accountability, and collaboration, have also been widely accepted in our policy documents. The benefit of transparency goes beyond creating accountability; it is also a core part of working on public services. The public should be able to see the work that the government is doing for their benefit. Transparency makes it clear that progress is being made on important services and issues and keeps the public informed of future plans. Open sourced work encourages active feedback and participation from the public, who can now view the progress as it happens and hold more confidence and trust in the work. With the exception of a small number of cases, software for public benefit should always be open source.

To make open source software functional, governments have to licence their code appropriately and make their code discoverable and available publicly. Software paid for by the public should (except in certain circumstances) be hosted in public view. Furthermore, some of the benefits of making open source code – such as enabling reuse, providing transparency and accountability to algorithms, and facilitating contributions from the public – are impossible without the code being available publicly online.

In order to have a truly open government, government employees need to be empowered with a better understanding of and appreciation for the virtues of open source. Departments and agencies who have open sourced code without any pressure from civil society often do so because of individual supporters spearheading the charge. Successful projects should be scoped from day one with the intention of being open source and serving to reshape market demand.

Given that the GoI's FOSS policies specifically call for it, government contractors and vendors have a duty to explore open source alternatives and to educate the market on modern industry-standard development practices. Creating internal competencies is also essential to grow government demand for FOSS. The government must be more explicit in its use of the term 'open'. Abstract definitions that are open to interpretation and left at the discretion of administrative functionaries to implement are not really open. A fully evolved form of the e-state involves citizen participation,[73] not just as end-users but also as informed stakeholders in the democratic process of governance.

[73] "Promoting e-Governance: The SMART Way Forward", Second Administrative Reforms Commission, Government of India, Eleventh Report, 2008, https://darpg.gov.in/sites/default/files/promoting_egov11.pdf

Appendix 1: Factors impacting the community health of a FOSS project

Some of the factors that are important to consider in determining the real health and openness of the Open Source[74] GovTech project are:

- Critical mass: A large enough userbase to sustain the project.
- Issue tracking: A clear issue tracking system that allows everyone to report and track the status of bugs, features, and tasks.
- User advocacy: Users discuss the software with other potential users.
- Open forum: An open forum for participation.
- Outside participation: A key check on the health and life of the project is whether those outside the primary organisation are actually submitting patches.
- Site navigation leads quickly to code and distribution: It should be easy to find licensing terms and where to download the code on the website.
- Upstream patches: Much of the open source software is built from other open source software. A key community health factor of the whole is whether contributions can flow back up from the project to the packages it contains.
- Every single bit of your 'thing' is open source: Having 'pro' versions that are not open source means advanced features do not benefit from community effects.
- Release plan: Users know what the plan for future releases is.
- Granular participation: The ability to participate without joining a committee or an extremely active mailing list

[74] https://blog.opensource.org/how-open-source-is-your-open-source/

Appendix 2: Criteria for open source software according to the Open Source Initiative

Principle	Criteria
Free redistribution	The licence does restrict any party from selling or giving away the software as a component of an aggregate software distribution that contains programmes from several different sources.
	The licence does not require a royalty or other fee for such a sale.
	The programme includes the source code.
Source code	The programme includes the source code.
	The programme allows distribution in source code.
	The programme allows distribution in compiled form.
	If some form of a product is not distributed with its source code, there is a well-publicised means of obtaining the source code for no more than a reasonable reproduction cost (preferably downloading via the internet without charge).
	The source code is the preferred form in which the programmer would modify the programme.
	The source code is not deliberately obfuscated.
Derived works	The license allows modifications and derived works.
	The license allows modifications and derived works to be distributed under the same terms as the license of the original software.
Integrity of the author's source code	The licence allows the distribution of 'patch files' with the source code for the purpose of modifying the programme at build time.

	The licence explicitly permits the distribution of software built from modified source code.
	The licence allows derived works to carry a different name or version number from the original software.
No discrimination against persons or groups	The licence must not discriminate against any person or group of persons.
No discrimination against fields of endeavour	The licence does not restrict anyone from making use of the programme in a specific field of endeavour.
Distribution of license	The rights attached to the programme apply to all to whom the programme is redistributed without the need for the execution of an additional licence by those parties.
Licence must not be specific to a product	The rights attached to the programme do not depend on the programme being part of a particular software distribution.
	Parties to whom the programme is redistributed have the same rights as those that are granted in conjunction with the original software distribution (if the programme is extracted from that distribution and used or distributed within the terms of the programme's license).
License must not restrict other software	The licence does not place restrictions on other software that is distributed along with the licensed software.
License must be technology-neutral	No provision of the licence shall be predicated on any individual technology or style of interface.