Department of Electronics and Information Technology
Ministry of Communications and Information Technology
Government of India

सत्यमेव जयते

Digital India
Power To Empower

e-Governance Policy Initiatives under Digital India

The e-Kranti Framework

Policy on Adoption of Open Source Software for Government of India

Policy on Open Application Programming Interfaces (APIs) for Government of India

Framework for Adoption of Open Source Software in e-Governance Systems

Policy on Collaborative Application Development by Opening the Source Code of Government Applications

Policy on Use of IT Resources of Government of India

Application Development & Re-Engineering Guidelines for Cloud Ready Applications

E-mail Policy of Government of India

e-Governance Policy Initiatives under Digital India

सत्यमेव जयते

Department of Electronics and Information Technology
Ministry of Communications and Information Technology
Government of India

“e-Governance is easy governance, effective governance and economical governance.”

Shri Narendra Modi
Hon’ble Prime Minister

सत्यमेव जयते

मंत्री
संचार एवं सूचना प्रौद्योगिकी
भारत सरकार
MINISTER
COMMUNICATIONS & IT
GOVERNMENT OF INDIA

रवि शंकर प्रसाद
RAVI SHANKAR PRASAD

26th May, 2015

MESSAGE

Government of India accords highest priority to the Digital India programme. The implementation of e-Kranti, an integral component of Digital India, aims at "Transforming e-Governance for Transforming Governance" and is vital for the delivery of e-governance, easy governance and good governance in the country.

Hon'ble Prime Minister has emphasised on more than one occasion that e-governance is easy governance, effective governance and economical governance. The recent rapid advances in ICT have made e-governance a very potent tool for ushering in an era of good governance. The implementation of e-Governance projects like MyGov platform, Jeevan Pramaan, Wi-Fi Hotspots, BPOs in rural areas, e-commerce through post offices etc is just the start of silent electronic revolution that aims on delivering the Government services to citizens and businesses in electronic mode thereby fostering an ecosystem which would ensure strong and digitally inclusive and equitable development —all of which are the very ethos and purpose of e-governance. These e-governance projects need necessary support from the Government through policies, guidelines and frameworks.

Department of Electronics and Information Technology (DeitY) has recently taken several policy initiatives in the e-Governance domain that include, inter alia, e-Kranti, Open Source Software, Open APIs, E-mail Policy, Use of IT Resources, Collaborative Application Development and Application Development & Re-Engineering for Cloud Ready Applications. These policy initiatives would truly support all Central Ministries/ Departments as well as all States/UTs in leveraging the emerging technologies, making use of newer business models and revamping of existing projects so as to deliver the services electronically to citizens in an efficient, transparent and affordable manner.

I am happy to present the compendium on "e-Governance Policy Initiatives" as a ready reckoner and am sure that Government Departments, e-Governance domain experts and practitioners would make the best use of it in implementing their projects.

I also congratulate all the officers and staff of DeitY involved in this endeavour for coming up with this compendium at the right moment when projects under Digital India and e-Kranti are being implemented across the country.

(RAVI SHANKAR PRASAD)

Overview

The Department of Electronics and Information Technology (DeitY), Ministry of Communications and Information Technology has taken several policy initiatives in the e-Governance domain that are crucial for achieving the vision and objectives of the Digital India programme. Effective implementation of e-Governance is a key component of the Digital India programme. These policy initiatives are an endeavor to chart out the roadmap for implementation of e-Governance projects in the country. They cover a number of important areas, e.g. e-Kranti (National e-Governance Plan 2.0), open source software, open APIs, e-mail policy, use of IT Resources, Collaborative Application Development and Application Development & Re-Engineering for Cloud Ready Applications. These policies are envisaged to provide necessary support to all Central Ministries/ Departments as well as all States/UTs in leveraging the emerging technologies, making use of newer business models and revamping of existing projects so as to deliver the services electronically to citizens in an efficient, transparent and affordable manner. These policies draw their strengths from the national and international best practices in the respective domain as well as inputs from subject matter experts from Government departments, industry and academia.

"The e-Kranti Framework" (chapter 1) provides details on the e-Kranti framework that is an integral part of the Digital India programme. With the vision of "Transforming e-Governance for Transforming Governance", e-Krantiprogramme aims towards easy governance, effective governance, good governance and mobile governance. It provides the key principles for revamping the existing projects and also for new and ongoing e-Governance projects.

The "Policy on Adoption of Open Source Software for Government of India" (chapter 2) will encourage the formal adoption and use of Open

Source Software (OSS) in Government organizations. The compliance to this policy will ensure that strategic control of e-Governance assets would remain with the Government and would also ensure business continuity for the projects in future from technical perspective.

The "Framework for Adoption of Open Source Software in e-Governance Systems" (chapter 3) suggests a set of recommendations and procedures for promoting, managing and enhancing the adoption of OSS in e-Governance Systems in India. It highlights the impact of adoption of OSS in Government, influencing factors, mutual impact of Open Standards and OSS, establishing enterprise security with OSS, unified software development for all major devices using standards based web browser and use of localisation. The Framework suggests neutral guidelines to select software and the process for induction of OSS solution. The ecosystem suggested to promote the adoption of OSS describes required institutional mechanism, collaboration with key stakeholders like industry, OSS communities, academia, collaborative mechanism, offering of services based on OSS, provisioning of support services on OSS and integration with on-going initiatives.

The "Policy on Open Application Programming Interfaces (APIs) for Government of India" (chapter 4) sets out the Government's approach on the use of "Open APIs" to promote software interoperability for all e-Governance applications & systems and provide access to data & services for promoting participation of all stakeholders including citizens. This policy initiative will encourage the formal use of Open APIs in Government organizations. The world-wide initiatives on "Open Government" also focus on open APIs to easily access the information collected by Government organizations.

The "E-mail Policy of Government of India" (chapter 5) lays down the guidelines with respect to use of e-mail services by the Government departments and organizations. The policy initiative aims to ensure secure access and usage of Government of India e-mail services by its users and is applicable to all employees of Government of India (GoI) and employees of those State/UT Governments that use the e-mail services of GoI.

The "Policy on Use of IT Resources of Government of India" (chapter 6) provides the guidelines to ensure proper access to and usage of Government's IT resources and prevent their misuse by the users. The policy initiative covers all IT resources including desktop devices, portable and mobile devices, networks including wireless networks, Internet connectivity, external storage devices and peripherals like printers and scanners and the software associated therewith.

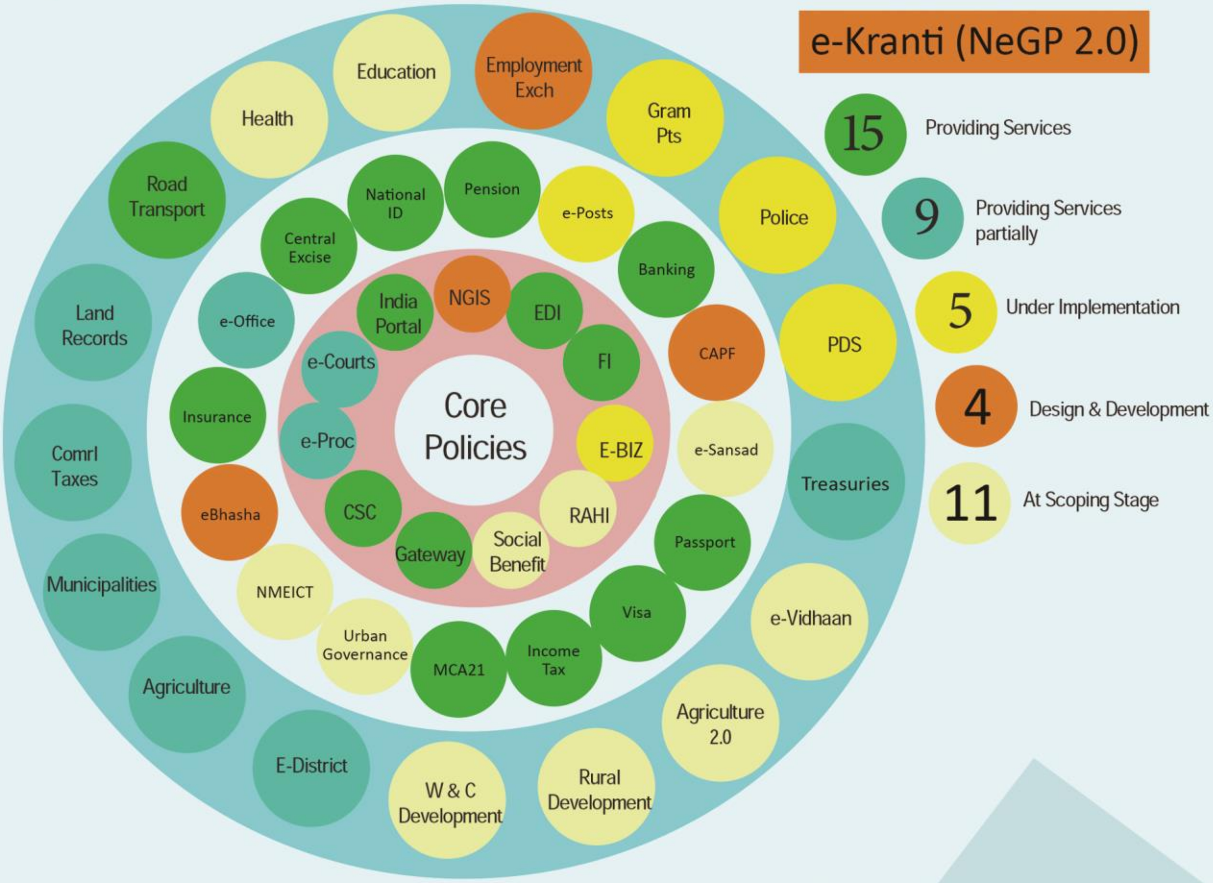
The "Policy on Collaborative Application Development by Opening the Source Code of Government Applications" (chapter 7) intends to increase the pace of e-Governance application development and rapid roll out/implementation by adopting an open-source based development model. The Government of India wants to promote re-use of existing developed applications. By opening the source code, the Government wants successful, scalable, high quality e-governance applications to be developed in a collaborative manner. It also wants new applications to be developed to encourage creativity both inside and outside the Government by encouraging collaborative development betweenGovernment departments/agencies and private organizations, citizens and developers to create innovative e-Governance applications and solutions.

The "Application Development & Re-Engineering Guidelines for Cloud Ready Applications" (chapter 8) aims to address one of the major concerns in the e-Governance domain that is lack of process reengineering and leveraging of the latest emerging technology i.e. Cloud. This guideline intends to ensure development of Common Application Software (CAS) which can be configured as per different States / departments requirements without the need of modifying the core code of the application for a faster deployment so that time, effort and cost in developing applications are saved and to avoid duplication of efforts. It is therefore imperative that applications are developed in conformity to guidelines that makes them standardized and compatible for hosting and running across states.

Table of Contents

Contents

The e-Kranti Framework



e-Kranti (NeGP 2.0)

15 Providing Services

9 Providing Services partially

5 Under Implementation

4 Design & Development

11 At Scoping Stage

Core Policies

India Portal

NGIS

EDI

FI

E-BIZ

RAHI

Social Benefit

Gateway

CSC

e-Proc

e-Courts

Central Excise

National ID

Pension

e-Posts

Banking

CAPF

e-Sansad

Passport

Visa

Income Tax

MCA21

Urban Governance

NMEICT

eBhasha

Insurance

e-Office

Road Transport

Health

Education

Employment Exch

Gram Pts

Police

PDS

Treasuries

e-Vidhaan

Agriculture 2.0

Rural Development

W & C Development

E-District

Agriculture

Municipalities

Comrl Taxes

Land Records

The e-Kranti Framework

Digital India programme aims at transforming India into a digitally empowered society and knowledge economy. The implementation of e-Kranti, an integral part of Digital India, is vital for e-Governance in the country.All the new Mission Mode Projects (MMPs) are required to follow the key principles of e-Kranti, namely ‘Transformation and not Translation’, ‘Integrated Services and not Individual Services’, ‘Government Process Reengineering (GPR) to be mandatory in every MMP’, ‘ICT Infrastructure on Demand’, ‘Cloud by Default’, ‘Mobile First’, ‘Fast Tracking Approvals’, ‘Mandating Standards and Protocols’, ‘Language Localization’, ‘National GIS (Geo-Spatial Information System)’, ‘Security and Electronic Data Preservation’. All the existing MMPs would be revamped in accordance with the aforesaid principles of e-Kranti.

Considering the relevance and impact of e-Kranti on all Government Ministries / Departments and involvement of multiple implementing agencies, it has been decided that the overall responsibility for each component of e-Kranti will be with respective domain Ministry / Department.

The “e-Kranti Framework” provides the following details:

- Role of e-Kranti in Digital India
- Objectives of e-Kranti
- Principles of e-Kranti
- Approach and Methodology
- Implementation Strategy
- Key Components

Chapter 1: The e-Kranti Framework

1.1 Preamble

Government of India accords the highest priority to the Digital India programme that is an umbrella programme for transforming India into a digitally empowered society and knowledge economy. e-Kranti is an integral part of the Digital India programme with the vision of "Transforming e-Governance for Transforming Governance". The mission of e-Krantis "To ensure a Government wide transformation by delivering all Government services electronically to the citizens through integrated and interoperable systems via multiple modes while ensuring efficiency, transparency and reliability of such services at affordable costs."

1.2 Role of e-Kranti in Digital India and its approval

The Union Cabinet in its meeting held on 25.03.2015 has approved the Approach and Key Components of e-Kranti that includes, inter alia, the vision, mission, key principles of e-Kranti, Approach and Methodology, Programme Management Structure and Implementation Strategy including 44 Mission Mode Projects and Core ICT Infrastructure. The Digital India programme and specifically its pillar 4and pillar 5 namely '**e-Governance: Reforming Government through Technology**' and '**e-Kranti - Electronic Delivery of Services**' respectively are directly linked with **e-Kranti** and the implementation of e-Kranti is critical for the success of e-governance, easy governance and good governance in the country.

What is thisframework?

- Overarching framework for the implementation of e-Governance projects
- Implementation of e-Kranti to ensure that e-Governance projects deliver outcome based services to citizens, businesses and also to Government
- 44 Mission Mode Projects (MMPs) to deliver various domain specific services
- Core ICT Infrastructure to provide front end, backend and middleware support
- Capacity building

1.3 Objectives of e-Kranti

The **objectives** of 'e-Kranti' are as follows:

i. To redefine NeGP with transformational and outcome oriented e-Governance initiatives

ii. To enhance the portfolio of citizen centric services

iii. To ensure optimum usage of core Information & Communication Technology (ICT)

iv. To promote rapid replication and integration of e-Governance applications

v. To leverage emerging technologies

vi. To make use of more agile implementation models

1.4 Principles of e-Kranti

The key principles of e-Kranti are as follow:

i. **Transformation and not Translation** - All project proposals in e-Kranti must involve substantial transformation in the quality, quantity and manner of delivery of services and significant enhancement in productivity and competitiveness.

ii. **Integrated Services and not Individual Services** - A common middleware and integration of the back end processes and processing systems is required to facilitate integrated service delivery to citizens.

iii. **Government Process Reengineering (GPR) to be mandatory in every MMP** - To mandate GPR as the essential first step in all new MMPs without which a project may not be sanctioned. The degree of GPR should be assessed and enhanced for the existing MMPs.

iv. **ICT Infrastructure on Demand** – Government departments should be provided with ICT infrastructure, such as connectivity, cloud and mobile platform on demand. In this regard, National Information Infrastructure (NII), which is at an advanced stage of project formulation, would be fast-tracked by DeitY.

v. **Cloud by Default** - The flexibility, agility and cost effectiveness offered by cloud technologies would be fully leveraged while designing and hosting applications. Government Cloud shall be the default cloud for Government Departments. All sensitive information of Government Departments shall be stored in a Government Cloud only. Any Government Department may use a private cloud only after obtaining permission from Department of Electronics and Information Technology which shall do so after assessing the security and privacy aspects of the proposed cloud.

vi. **Mobile First** - All applications are designed/ redesigned to enable delivery of services through mobile.

vii. **Fast Tracking Approvals** – To establish a fast-track approval mechanism for MMPs, once the Detailed Project Report (DPR) of a project is approved by the Competent Authority, empowered committees may be constituted with delegated powers to take all subsequent decisions.

viii. **Mandating Standards and Protocols** – Use of e-Governance standards and protocols as notified by DeitYbe mandated in all e-governance projects.

ix. **Language Localization** - It is imperative that all information and services in e-Governance projects are available in Indian languages as well.

x. **National GIS (Geo-Spatial Information System)** - NGIS to be leveraged as a platform and as a service in e-Governance projects.

xi. **Security and Electronic Data Preservation** - All online applications and e-services to adhere to prescribed security measures including cyber security. The National Cyber Security Policy 2013 notified by DeitY must be followed.

1.5 Approach and methodology for implementing e-Kranti

The following Approach and Methodology should be adopted for e-Kranti:

i. Ministries / Departments / States would fully leverage the Common and Support ICT Infrastructure (e.g. GI Cloud, National / State Data Centres, Mobile Seva, State Wide Area Networks, Common Services Centres& Electronic Service Delivery Gateways). DeitY would also evolve/ lay down standards and policy guidelines, provide technical and handholding support, undertake capacity building, R&D, etc.

ii. The existing/ ongoing MMPs would also be suitably revamped to align them with the principles of e-Kranti. Scope enhancement, Process Reengineering, use of integrated & interoperable systems and deployment of emerging technologies like cloud & mobile would be

Why weneed it?

- Existing e-Governance projects, by and large, lack necessary process reengineering, use of emerging technologies like cloud and mobile, use of metadata and data standards, etc
- At present, there are numerous solutions but they are mostly suboptimal in performance
- e-Kranti aims to address the aforesaid issues in the e-Governance projects

undertaken to enhance the delivery of government services to citizens.

iii. States would be given flexibility to identify, for inclusion, additional state-specific projects, which are relevant for their socio-economic needs.

iv. e-Governance would be promoted through a centralised initiative to the extent necessary, to ensure citizen service orientation, interoperability of various e-Governance applications and optimal utilisation of ICT infrastructure/ resources, while adopting a decentralised implementation model.

v. Successes would be identified and their replication promoted proactively with required customisation wherever needed.

vi. Public Private Partnerships would be preferred wherever feasible to implement e-Governance projects with adequate management and strategic control.

vii. Adoption of Aadhaar based ID would be promoted to facilitate identification and delivery of benefits.

1.6 Implementation Strategy of e-Kranti

For implementation of the e-Kranti, various Central Ministries/ Departments and State Governments would be involved. Considering the multiplicity of agencies involved and the need for overall aggregation and integration at the national level, it is considered appropriate to implement e-Kranti as a programme, with well defined roles & responsibilities of each agency involved, and to create an appropriate programme management structure.

For the e-Kranti, following role assignments/ responsibilities are being followed/ proposed:

(a) The proposed Apex Committee on Digital India programme, constituted with Cabinet Secretary as its Chairman and Secretary, DeitY as its Member Convener, would be overseeing the e-Krantiprogramme also and providing policy and strategic directions for its implementation and resolving inter-ministerial issues. The Apex Committee, in addition would harmonize and integrate diverse initiative aspects related to integration of services, end to end process re-engineering and service levels of MMPs wherever required.

(b) Line Ministries/Departments would be responsible for the implementation of the assigned Mission Mode Projects (MMPs)/Components as indicated in **Annexure**. Mission Mode Projects would be owned and spearheaded by various line Ministries for Central Government, State Governments and Integrated projects **Annexure**. Each Department would work in a project mode within a tight, defined timeframe by preparing a detailed project document, either in-house or with the assistance of a Consultant. This document should clearly spell out all important aspects of the project like services and service levels, project implementation team, process reengineering proposed, change management plan, project management plan, timelines, etc. The services and service levels would be determined in consultation with the actual users and for this, each concerned department would form an Advisory Committee, on which users would also be represented.

(c) State Governments would be responsible for implementing State Sector MMPs, under the overall guidance of respective Line Ministries in cases where Central Assistance is also required. An Apex Committee on Digital India proposed to be constituted at the State level headed by the Chief Secretary would be used to monitor the e-Kranti implementation at state level. They would also analyse State specific requirements and recommend project proposals for inclusions/ deletions from the listed MMPs.

(d) DeitY would be the facilitator and catalyst for the implementation of e-Kranti by various Ministries and State Governments and would also provide technical assistance to them either directly or in collaboration with external professional Consultants. It would serve as a secretariat to the Apex Committee and assist it in managing the programme. In addition, it would implement pilot/ infrastructure/ technical/ special projects and support components including those indicated in **Annexure**. DeitY would also prepare a suitable template for preparing project document, which could be used by individual departments for preparing their detailed project reports.

(e) DAR&PG would continue its responsibility towards Generic Process Re-engineering and Change Management, which are desired to be realised across all government departments. However, to upscale NeGP to deliver services, DAR&PG would focus on transformational approach in the Government Process Re-engineering (GPR) initiatives of various Ministries / Departments. For various Mission Mode Projects, concerned Line Ministries/ Implementing Agencies would be primarily responsible for carrying out the required Process Re-engineering and Change Management. DAR&PG/ DeitY would also be promoting initiatives for Human Resource Development and Training and Awareness building.

(f) Planning Commission and Ministry of Finance would allocate funds for implementing e-Kranti both in existing and new MMPs through Plan and Non-plan budgetary provisions and lay down appropriate procedures in this regard. The projects in the portfolio of e-Kranti should be exempted from all budgetary restrictions and cuts such that the projects get implemented in time.

(g) Once the DPR of a project is approved by the Competent Authority, the Empowered Committee constituted for the purpose would be truly empowered to take all subsequent decisions, which should be implemented soon after the minutes of the EC are approved.

(h) The Council of Mission Leaders for Digital India proposed as a platform to share the best practices in Mission Mode Projects under NeGP and new e-Governance initiatives of DeitY would perform its envisaged role and responsibilities.

(i) The inter-departmental, integration and interoperable issues of integrated projects / e-Governance initiatives would be resolved by the Apex Committee headed by Cabinet Secretary. And the technical issues of integrated projects would be resolved by the Council of Mission Leaders headed by Secretary, DeitY.

How will it be implemented?

- New Mission Mode Projects (MMPs) will follow the key principles of e-Kranti
- Existing e-Governance projects / initiatives will be revamped in accordance with the principles of e-Kranti
- Responsibility of each component of e-Kranti will be with respective domain Ministry / Department
- Financial details will be worked out project-wise by the Line Ministries/ Departments/ State Governments concerned
- Programme management structure institutionalized at both national and State/UT level

1.7 Annexure – Key Components of e-Kranti

The National E-Governance Plan (NeGP) was first conceived in mid 2003, by the D/o Electronics and Information Technology (DeitY) and the D/o Administrative Reform & Public Grievances (DAR&PG) and received in-principle approval at the level of the then Prime Minister on the 6th November 2003. Subsequently, Cabinet Secretary took follow up meetings of the Core Group on **Administrative** Reforms as well as of the Committee of Secretaries on 14.11.2003 wherein 22 Mission Mode Projects were identified for implementation on a priority basis. Four more projects have been added to the list of Mission Mode Projects namely **e-Courts** on the suggestion of the Judiciary, **e-Office** on the suggestion of DAR&PG, **e-Procurement** on the suggestion of CVC, and **Employment Exchanges** at the instance of the Planning commission. Thereafter, Apex Committee on NeGP headed by the Cabinet Secretary reviewed the progress of NeGP and accorded in principle approval to add 4 MMPs namely **Education, Health, PDS** and **Posts** under the MMP portfolio of NeGP on

29th July, 2011. Subsequent to the conceptualization of National e-Governance Plan 2.0 (NeGP 2.0), the 10 MMPs namely **e-Sansad, e-Vidhaan, Financial Inclusion, Roads and Highways Information System (RAHI), Agriculture 2.0, National Geographical Information System (NGIS), Rural Development, Social Benefits, Women and Child Development and Common IT Roadmap for Para Military Forces** are accorded in principle approval by the Apex Committee on NeGP headed by Cabinet Secretary on 18th March, 2014. e-Bhasha, Urban Governance and National Mission on Education Through ICT (NMEICT) are proposed as new MMPs under Integrated Services Category.

The e-Kranti now covers 44 Mission Mode Projects in three categories: Central, States and Integrated Services. Details of these Mission Mode Projects are given in the Tables I to III below. Some of these projects are under various stages of implementation and may require some transformational process reengineering, refinements and adjustment of scoping and implementation strategy to achieve the desired service level objectives by the concerned line Ministries/Departments at the Central, State and Local Government levels. All these Mission Mode Projects have the common aim of improving delivery of Government services to citizens and businesses.

Table-I: Mission Mode Projects Central Government Category

S. N.	Project	Line Ministry/ Department Responsible
01	Income Tax	M/o Finance/Central Board of Direct Tax
2	Passport	M/o External Affairs
03	MCA21	M/o Company Affairs
04	Insurance	D/o Financial Services
05	National Citizen Database	M/o Home Affairs/Registrar General of India (RGI)
06	Central Excise	D/o Revenue/Central Board of Excise & Custom
07	Pensions	D/o Pensions & Pensioners welfare & Dept. of Expenditure
08	Banking	D/o Financial Services
09	e-Office	D/o Administrative Reforms & Public Grievances
10	Posts	D/o Posts
11	Visa & Immigration	M/o Home Affairs
12	e-Sansad#	Parliament of India, Lok-Sabha Secretariat
13	Common IT Roadmap for Para Military Forces#	M/o Home affairs

# These MMPs are New MMPs under e-Kranti.

Table-II: Mission Mode Projects State Government Category

S. N.	Project	Line Ministry/ Department Responsible
01	Land Records	M/o Rural Development
02	Road Transport	M/o Road Transport & Highway
03	Property Registration	D/o Land Resources and D/o Electronics and Information Technology
04	Agriculture	D/o Agriculture & Cooperation
05	Treasuries	M/o Finance
06	Municipalities	M/o Urban Development and Poverty Alleviation
07	Gram Panchayats	M/o Panchayati Raj
08	Commercial Taxes	M/o Finance
09	Police	M/o Home affairs
10	Employment Exchanges	M/o Labour & Employment
11	School Education	D/o School Education and Literacy
12	Health	D/o Health and Family Welfare
13	PDS	D/o Food and Public Distribution
14	e-Vidhaan#	Parliament of India, Lok-Sabha Secretariat
15	Agriculture 2.0#	D/o Agriculture
16	Rural Development#	D/o Rural Development
17	Women and Child Development#	M/o Women and Child Development

# These MMPs are New MMPs under e-Kranti.

Table-III: Mission Mode Projects Integrated Services Category

S. N.	Project	Line Ministry/ Department Responsible
01	EDI (E-Commerce)	M/o Commerce & Industry and D/o Commerce
02	E-Biz	D/o Industrial Policy & Promotion and D/o Electronics and Information Technology
03	Common Services Centres	D/o Electronics and Information Technology
04	India Portal	D/o Electronics and Information Technology and D/o Administrative Reforms & Public Grievances
05	E-Courts	D/o Justice, M/o Home Affairs
06	E-Procurement	M/o Commerce & Industry/ DGS&D
07	National Service Delivery Gateway	D/o Electronics and Information Technology
08	Financial Inclusion#	D/o Financial Services
09	National Geographical Information System#	D/o Science & Technology
10	Social Benefits#	M/o Social Justice and Empowerment as the leader and other welfare departments as co-owners
11	Roads and Highways Information System (RAHI) #	M/o Road Transport & Highways
12	e-Bhasha #	D/o Electronics and Information Technology
13	National Mission on Education Through ICT (NMEICT) #	D/o Higher Education
14	Urban Governance #	Ministry of Urban Development

# These MMPs are New MMPs under e-Kranti.

2. The thrust areas of e-Kranti outlined under Digital India programme are as follows:

Table-IV: Thrust areas and sub components of e-Kranti outlined in Digital India

S.N.	Areas	Sub components
1	**Technology for Education (e-Education)**	• All Schools connected with broadband • Free wifi in all schools (250,000) • Digital Literacy program • MOOCs – develop pilot Massive Online Open Courses
2	**Technology for Health (e-Healthcare)**	• Online medical consultation • Online medical records • Online medicine supply • Pan-India exchange for patient information • Pilots – 2015; Full coverage in 3 years
3	**Technology for Planning**	• GIS based decision making • National GIS MMP
4	**Technology for Farmers**	• Real time price information • Online ordering of inputs • Online cash, loan, relief payment with mobile banking
5	**Technology for Security**	• Mobile Emergency Services
6	**Technology for Financial Inclusion**	• Mobile Banking • Micro-ATM program • CSCs/ Post Offices
7	**Technology for Justice**	• e-Courts, e-Police, e-Jails, e-Prosecution
8	**Technology for Cyber Security**	• National Cyber Security Co-ordination Center

* Ongoing Mission Mode Projects under NeGP will be revamped to cover aforesaid areas and its sub components outlined in Digital India programme.

3. e-Governance: Reforming Government through Technology is one amongst the nine pivotal pillars of the Digital India Programme. Its major components are as follows:

Table-V: Components and sub components for Reforming Government through Technology under Digital India

S.N.	Major Components	Content
1	Government **Business Process Re-engineering** using IT to improve transactions	• Form Simplification, reduction • Online applications and tracking, Interface between departments • Use of online repositories e.g. school certificates, voter ID cards, etc. • Integration of services and platforms – UIDAI, Payment Gateway, Mobile Platform, EDI
2	**Electronic Databases**	• All databases and information to be made electronic, not manual
3	**Workflow automation**	• Workflow inside government offices to be made automated and visible to citizens
4	**Public Grievance Redressal using IT**	• Using IT to automate, respond, analyse data to identify and resolve persistent problems • Largely process improvements

*The critical transformational components would be implemented across the government Ministries / Departments.

4. To sustain the above projects there is also a need to create the right governance and institutional mechanisms, set up core infrastructure, formulate key policies, standards and the legal framework for adoption and to channelise private sector technical and financial resources into the National e-Governance efforts. For this purpose, certain key components have also been identified for implementation and the same are given in Table VI below. These components cut across and support various projects.

Table-VI: Support Components Category

Sl. No.	Support Components	Line Ministry/ Department Responsible
01	Core Policies (Cyber Security Policy, National IT Policy, Open Standard Policy etc.)	D/o Electronics and Information Technology
02	Core Infrastructure (SWAN, NII, SDCs, Mobile Seva, Payment Gateway, GI Cloud etc.) *	D/o Electronics and Information Technology
03	Support Infrastructure (CSCs, etc.) *	D/o Electronics and Information Technology
04	Technical Assistance	D/o Electronics and Information Technology
05	R&D	D/o Electronics and Information Technology
06	Human Resource Development & Training	D/o Electronics and Information Technology and D/o Administrative Reforms & Public Grievances
07	Awareness & Assessment	D/o Electronics and Information Technology and D/o Administrative Reforms & Public Grievances
08	Organization structures	D/o Electronics and Information Technology and D/o Administrative Reforms & Public Grievances

* **SWAN**: State Wide Area Network, **NII**: National Information Infrastructure, **SDC**: State Data Centre, **CSCs**: Common Services Centres, **GI Cloud** – Government of India Cloud (MeghRaj)

With advancement in technologies like mobile, cloud, data analytics andsocial media and emergence of new business models like infrastructure on demand model, plug and playmodel and outcome based / transaction based charging, many new opportunities have appeared in the horizon, whichwere neither availablenorpracticalearlier. The scale, scope and speed of information exchange and data transfer has increased manifold in recent times and these require that Government's decision making and service delivery should be adequate and fastto meet the need and aspirations of the common citizens in the 21st century. The e-Kranti framework addresses the electronic delivery of services through a portfolio of mission mode projects that cut across several Government Department. It also covers essential requirements of Core ICT Infrastructure that include, inter-alia, GI Cloud, Data Centre, network connectivity, common platforms like Aadhaar, Mobile Seva, Payment Gateway, etc.

Thekey principle of e-Kranti namely'Integrated Services and not Individual Services', 'Mandatory Government Process Reengineering (GPR) in every MMP', 'ICT Infrastructure on Demand', 'Cloud by Default', 'Mobile First', etc, would ensure the realization of vision of e-Kranti i.e. "Transforming e-Governance for Transforming Governance".

Policy on Adoption of Open Source Software

Policy on Adoption of Open Source Software

Digital India aims to make Government services digitally accessible to citizens in their localities and to ensure efficiency, transparency and reliability of such services at affordable costs. Government of India endeavours to adopt Open Source Software in all e-Governance systems implemented by various Government organizations as a preferred option.The policy shall be applicable to all Government organisations under the Central Governments and those State Governments that choose to adopt this policy for e-Governance systems.

This policy provides details on the following:

- Objectives
- Nature of compliance
- Applicability
- How to Comply
- Exception
- Implementation Mechanism

Chapter 2: Policy on Adoption of Open Source Software

2.1 Preamble

Government of India (GoI) is implementing the Digital India programme as an umbrella programme to prepare India for a knowledge based transformation into a digitally empowered society and a knowledge economy. Under the overarching vision of Digital India, GoI aims to make Government services digitally accessible to citizens in their localities and to ensure efficiency, transparency and reliability of such services at affordable costs. To meet this objective, there is a need to set up a commensurate hardware and software infrastructure, which may require significant resources.

Organizations worldwide have adopted innovative alternative solutions in order to optimise costs by exploring avenues of "Open Source Software". GoI has also been promoting the use of open source technologies in the e-Governance domain within the country in order to leverage economic and strategic benefits.

Further, the National Policy on Information Technology, 2012 has mentioned, as one of its objectives, to "Adopt open standards and promote open source and open technologies".

In view of the above, there is a need to formulate a policy for the Government Organizations to adopt Open Source Software. The "Policy on Adoption of Open Source Software for Government of India" (hereinafter referred to as "Policy") will encourage the formal adoption and use of Open Source Software (OSS) in Government Organizations.

What is this policy?

- Policy provides a framework for rapid and effective adoption of OSS

2.2 Objectives

- To provide a policy framework for rapid and effective adoption of OSS
- To ensure strategic control in e-Governance applications and systems from a long-term perspective.
- To reduce the Total Cost of Ownership (TCO) of projects.

Why we need it?

- Use of open source technologies to leverage economic and strategic benefits
- Source code will be available to open source software community for study and modification
- Source code shall be free from any royalty

2.3 Policy Statement

Government of India shall endeavour to adopt Open Source Software in all e-Governance systems implemented by various Government organizations, as a preferred option in comparison to Closed Source Software (CSS).

The Open Source Software shall have the following characteristics:

- The source code shall be available for the community / adopter / end-user to study and modify the software and to redistribute copies of either the original or modified software.
- Source code shall be free from any royalty.

2.4 Nature of Compliance

Mandatory

2.5 Applicability

The policy shall be applicable to all Government Organisations under the Central Governments and those State Governments that choose to adopt this policy for the following categories of e-Governance systems:

- All new e-Governance applications and systems being considered for implementation.
- New versions of the legacy and existing systems.

2.6 How to comply

All Government Organizations, while implementing e-Governance applications and systems must include a specific requirement in Request for Proposal (RFP) for all suppliers to consider OSS along with CSS while responding. Suppliers shall provide justification for exclusion of OSS in their response, as the case may be. Government Organizations shall ensure compliance with this requirement and decide by comparing both OSS and CSS options with respect to capability, strategic control, scalability, security, life-time costs and support requirements.

2.7 Exception

GoI shall endeavour to adopt Open Source Software in all e-Governance applications and systems implemented by Government Organizations. However, in certain specialised domains where OSS solutions meeting essential functional requirements may not be available or in case of urgent / strategic need to deploy CSS based solutions or lack of expertise (skill set) in identified technologies, the concerned Government Organization may consider exceptions, with sufficient justification.

2.8 Implementation Mechanism

i) GoI shall publish a policy framework for rapid and effective adoption of OSS covering the prioritization of the application areas and illustrative list of OSS & OSS Stacks etc, required for various functional areas.

ii) All future Requests for Proposals (RFPs) of e-Governance projects shall include a mandatory clause for considering Open Source Software (OSS) as a preferred option in comparison to Closed Source Software (CSS). Suppliers shall provide justification for exclusion of OSS in their response.

iii) Government Organizations shall ensure compliance with this requirement and decide by comparing both OSS and CSS options with respect to capability, strategic control, scalability, security, life-time costs and support requirements.

iv) GoI shall establish suitable support mechanism for the available OSS that includes Institutional Mechanism, Partnership with Industry, Academia and OSS Community.

v) GoI shall actively collaborate with OSS communities in India as well as at the International level and contribute wherever appropriate.

2.9 Review of the Policy

GoI shall have the right to review the Policy as and when required.

2.10 Point of Contact

All queries or comments related to the “Policy on Adoption of Open Source Software for Government of India” shall be directed to JS (e-Governance),DeitY (jsegov@deity.gov.in).

How will it be implemented?

- GoI shall publish a policy framework for rapid and effective adoption of OSS
- RFPs of e-Governance projects shall have a clause on OSS as a preferred option
- Government organizations shall ensure compliance
- GoI shall establish suitable support mechanism for the available OSS
- GoI shall actively collaborate with OSS communities in India and abroad

The e-Governance projects involve development of applications and databases for the delivery of citizen centric services through web or mobile platforms. Such applications are developed through various technologies, which could be open source software or closed source software. The source code of open source software is available to the developer community, which can make the necessary changes in the software as per any changes in requirements. This advantage is not available to the closed source software as their source code is not available to the developer community. This aspect of closed source software could pose a hindrance in ensuring strategic control with the Government. This is the main reason why, Governments worldwide are trying to promote application development in open source software.

Framework for Adoption of Open Source Software in e-Governance Systems

Framework for Adoption of Open Source Software in e-Governance Systems

This framework suggests a set of recommendations and procedures for promoting, managing and enhancing the adoption of Open Source Software (OSS) in e-Governance Systems in India. The Framework suggests neutral guidelines to select software and the process for induction of an OSS solution. The ecosystem suggested to promote the adoption of OSS describes the required institutional mechanism, collaboration with key stakeholders like industry, OSS communities, academia, collaborative mechanism, offering of services based on OSS, provisioning of support services on OSS and integration with the on-going initiatives.

The framework provides the following details:

- Scope and Applicability
- OSS Current Scenario
- Factors influencing the adoption of OSS
- Impact of adoption of OSS
- Types of OSS support models
- OSS licenses
- Security aspects
- Unified Software Development
- Rapid Application Development
- Localisation and OSS
- Device driver
- Procurement guidelines
- Stages for induction of OSS solution
- Proposed ecosystem

Chapter 3: Framework for Adoption of Open Source Software in e-Governance Systems

3.1 Metadata

S. No.	Data elements	Values
1.	**Title**	Framework for Adoption of Open Source Software in e-Governance Systems
2.	**Title Alternative**	Framework for OSS Adoption
3.	**Document Identifier**	Framework for OSS Adoption: 01
4.	**Document Version, month, year of release**	Version 1.0 (April, 2015)
5.	**Present Status**	Final
6.	**Publisher**	Departmentof Electronics and InformationTechnology(DeitY), MinistryofCommunications&InformationTechnology (MCIT), GovernmentofIndia(GoI)
7.	**Date of Publishing**	April, 2015
8.	**Type of Standard Document**	Framework
9.	**Enforcement Category**	Advisory
10.	**Creator**	DeitY, NIC
11.	**Contributor**	OTC, NIC
12.	**Brief Description**	"Framework for Adoption of Open Source Software" suggests a set of recommendations and procedures for promoting, managing and enhancing the adoption of OSS in e-Governance Systems in India. It highlights the impact of adoption of OSS in Government, influencing factors, mutual impact of Open Standards and OSS, establishing enterprise security with OSS, unified software development for all major devices using standards based web browser and use of localisation.

S. No.	Data elements	Values
		The Framework suggests neutral guidelines to select software and the process for induction of OSS solution. The ecosystem suggested to promote the adoption of OSS describes required institutional mechanism, collaboration with key stakeholders like Industry, OSS Communities, Academia, collaborative mechanism, offering of services based on OSS, provisioning of support services on OSS and integration with on-going initiatives.
13.	**Target Audience**	• Government Departments and Agencies • Information and Communication Technology (ICT) industry (playing the roles of suppliers, developers, implementers and maintainers, integrators, service-providers) implementing e-Governance projects. • Academia working in e-Governance domain.
14.	**Owner of approved Framework**	DeitY,MCIT,NewDelhi
15.	**Subject**	Open Source Software
16.	**Subject. Category**	Adoption Framework
17.	**Coverage. Spatial**	INDIA
18.	**Format**	PDF
19.	**Language**	English
20.	**Rights. Copyrights**	DeitY, MCIT, New Delhi
21.	**Source**	Different resources, as indicated in the document
22.	**Relation**	Policy on Adoption of Open Source Software in GoI

3.2 Executive Summary

Government of India (GoI) is implementing the Digital India programme as an umbrella programme to prepare India for a knowledge based transformation into a digitally empowered society and a knowledge economy. Under the overarching vision of Digital India, GoI aims to make Government services digitally accessible to citizens in their localities and to ensure efficiency, transparency and reliability of such services at affordable costs. To meet this objective, there is a need to set up a commensurate hardware and software infrastructure, which may require significant resources.

Adoption of Open Source Software (OSS) has increased worldwide and has led to innovations in implementation of ICT solutions across businesses and Governments. The use of OSS in the key domains of ICT implementation (like application development, internet connectivity, infrastructure, Data Centre and mobile) has helped widespread adoption of open source technologies across the world. The OSS solutions have matured to a large extent and millions of committed developers are participating in making it conducive to the needs of different areas of ICT implementation. These solutions are now available with the required support services. The increased convergence of computing platforms facilitates the use of OSS together with Open Standards and adoption of web browser as a unified platform for software applications. The socio economic and strategic benefits offered by the adoption of OSS in e-Governance have encouraged several Governments and public agencies, to bring out policy framework / guidelines in this area. Compliance to Open Standards brings the twin benefits of interoperability and easy migration to OSS.

Government of India has been promoting the use of open source technologies and has been keenly encouraging their adoption in the e-Governance movement of the country. Department of Electronics and Information Technology (DeitY), Government of India has formulated The “Policy on Adoption of Open Source Software for Government of India” to enable effective adoption of OSS and encourage the formal adoption and use of Open Source Software (OSS) in Government Organizations. The policy has been approved and notified.

In pursuant to this policy, department is required to publish a policy framework for rapid and effective adoption of OSS covering the prioritization of the application areas and illustrative list of OSS and OSS stack etc. required for various functional areas. This "Framework for Adoption of Open Source Software" has been formulated to promote adoption of OSS in e-

Governance Systems in India. It lays down a set of recommendations and procedures for promoting, managing and enhancing the adoption of OSS.

The key objectives of the Framework are to:

(a) Provide guidance to the Govt. departments and agencies in selecting OSS Solutions

(b) Identify the OSS Stack appropriate to the needs of various government department and agencies

(c) Enhance & sustain the ecosystem to provide multi-layer support services on OSS for various National & State projects

(d) Create knowledge-base and build capacity on OSS

(e) Provisioning the Institutional Mechanism and resources required for promoting OSS on an ongoing basis.

3.3 Scope and Applicability

Scope	This Framework provides a set of recommendations and procedures for promoting, managing and adopting OSS as a preferred option in e-Governance Systems.
Applicability of this Document	All e-Governance systems.
Need for the Framework	(a)To implement one of the objectives of the National Policy on Information Technology, 2012 i.e. "Adopt Open standards and promote open source and open technologies”. (b)To widen the adoption of OSS to cover various National & State projects based on experience, expertise & feedback. (c)To imzprove the ecosystem of OSS (Support for OSS Tools, Project Planning, Development, Deployment, create community & industry support within the country and Capacity Building). (d)To minimise the informal use of OSS and absorption of OSS technology by limited number of internal experts (e)To mitigate the risks like hidden lock-ins and poor maintainability & sustainability of OSS. (f)To plan and provide the resources (time, funding, man-power and efforts) required to achieve the targets. (g)To reap the maximum socio economic benefits as a result of the adoption of OSS. (h)To improve citizen interface and similarly to establish systems for a better strategic control & ownership of e-Gov projects.
Targeted Stakeholders	(a)Government Departments and Agencies. (b)Information and Communication Technology (ICT) industry (playing the roles of suppliers, developers, implementers and maintainers, integrators, service-providers) implementing e-Governance projects. (c)Academia working in e-Governance domains.
When to use the framework	(a)Development & Implementation of new e-Governance systems. (b)Enhancements & Up-gradation of existing/legacy e-Governance systems.

Nature of Applicability	Advisory
Areas most suited for OSS:	1. Database, 2. Application/Web Server, 3. Server Operating System, 4. Software defined Networking, 5. Cloud Platform (including Virtualisation and Server Operating System), 6. Desktop Operating System (including Office Productivity Tool), 7. Cross-Platform Application Development (Unified Software Development for Mobile, Tablet, laptop and Desktop).

This Framework is prepared with a focus mainly on e-Governance Systems. However, other sectors can also use this Framework with benefit. A knowledge base on OSS will be created and shared under this Framework.

What is this framework?

- Provides a set of recommendations for promoting, managing and adopting OSS
- Helps Government departments and agencies in identifying and selecting OSS solutions

3.4 Overview of OSS

The software solutions developed by various business organisations and communities are deployed or released under various types of licenses and classified as Closed Source Software (CSS) / Proprietary Software, Shareware, Freeware and Open Source Software (OSS).

1. Closed Source Software / Proprietary Software

The conditions or license of CSS/proprietary software typically prohibit the access to / modification of the source code. It restricts the copy, modification, distribution and reuse of the software. The restrictions may be applicable to the whole or part of the software so that the control is with the concerned company. Revenue, profit and IPR drive the development and marketing of the products and solutions.

2. Shareware

The conditions of license of shareware are almost the same as the CSS license except that executables of the software are made available for restrictive-use free of charge for a specific trial-period.

3. Freeware

The conditions of license of freeware are almost the same as the shareware except that executables of the software are made available for restrictive-use free of charge permanently.

4. Open Source Software

OSS is also commonly known as Free and Open Source Software (FOSS), or Free Libre Open Source Software (FLOSS). Here the "Free" refers to "Freedom to use" and not "Free of Charge"; similarly, "Open Source" refers to the "Availability of Source code" for the community / adopter / end-user to study and modify the software and to redistribute copies of either the original or modified software (without having to pay royalties to previous developers). The definitions of Free Software and Open Source are made available by Free Software Foundation[1] and Open Source Initiative[2] respectively.

There are references which show the synergy between both FOSS & OSS; for example, the reference[3] shows

[1] Free Software Foundation, http://www.fsf.org

[2] Open Source Initiative (OSI), http://opensource.org

[3] Categories of Free and Nonfree Software - GNU Project - Free Software Foundation http://www.gnu.org/philosophy/categories.html

"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" and the site[4] says "They both refer to essentially the same thing".

Since, Open Standard and Open Hardware along with OSS/FOSS are also being adopted by many countries; the umbrella-term "Open Technology" is widely used. Based on these reasons the term, "Open Source Software" is adopted in this Framework. OSS has matured solutions at par with or better than CSS solutions.

5. Open Source Stack

There are varieties of OSS solutions available for each domain like Web Server, Database Server, Application Server, etc. Identifying, selecting and deploying the right solution is not a simple task. Project teams informally select and use arbitrarily chosen OSS solutions quite often, based on their preference and /or convenience. Such an informal usage of OSS solutions should be avoided to mitigate the risks like hidden lock-ins, poor maintainability of OSS, absorption of OSS technology by limited number of internal experts etc.

The OSS solution (component) needs to be identified for each domain through a set of guidelines. The identified OSS components are to be integrated, tested and offered as pre-configured OSS Stack. Such a stack is to be notified for adoption & reuse with support services in a formal way.

Most of the current e-Gov solutions are based on Java & PHP Technologies. Because of the proliferation of Mobiles & Tablets, Open Web Technology is included along with PHP & Java Technologies in the OSS Stacks listed in the Framework. These OSS Stacks are provisioned with an appropriate support model.

A typical OSS stack is given in <Annexure-I>. The OSS Tool-sets recommended should be given along with the support services at central, regional and state levels. As the Government has limited resources it is difficult to give support for all OSS Technologies chosen without appropriate criteria.

There is no bar in using other OSS Technologies in e-Governance; but the project team has to take care of the support for these technologies. If

[4] Debian -- What Does Free Mean? http://www.debian.org/intro/free

significant numbers of projects are based on other OSS Technologies then they would be considered in the OSS Stack in future.

3.5 Preamble

Declining Challenges for Adoption of OSS in Government

Many of the early barriers/ challenges to the use of OSS such as lack of awareness, lack of required skills and training, absence of appropriate business models, absence of standards and interoperability are rapidly reducing. As components of OSS mature, they become easier to use and maintain. A comprehensive list of popular OSS is given in <Annexure-II> "Illustrative List of OSS". Local firms, user community and developer community offer support and interoperability among different applications with obvious advantages. Alternative business models have emerged which allow OSS contributors to derive revenue for their efforts without charging for the software itself.

Impact of OSS in ICT and non-ICT Domains

OSS framework has a wider perspective than a software development methodology. It not only increases access, ownership and control of ICT, but also provides a Framework for usage and sharing of intellectual capital. The sharing of knowledge spreads, not only through OSS, but also through other related areas like Open Standards, Open Hardware and Product Designs, Open Process, Open ware Course, etc. This is collectively known as Open Technology (OT). In addition to ICT fields, the tradition of sharing of knowledge spreads in many other sectors as Open Medicine, Open Knowledge base, Open Law, Open Science, Open Music, Open Agriculture, etc.

SWOT Analysis of OSS Adoption

OSS adoption also provides many social, economic & strategic benefits described in terms of **S**trengths and **O**pportunities. At the same time, certain precautionary measures are required so as to realise maximum benefits. These measures are explained in terms of **W**eaknesses and **T**hreats. The **S**trengths, **W**eaknesses, **O**pportunities and **T**hreats (SWOT) Analysis of OSS Adoption is outlined below:

Strengths	Weaknesses
Include freedom to use & reuse, cost effectiveness, innovation, enhanced security, better local capacity building, preservation of foreign exchange, minimised piracy, better interoperability, community support, collaborative & distributed approach, enhanced competition, growing & mature developer ecosystems and rapid & effective vulnerability remediation.	Include adhoc use of OSS, adverse impact of legacy systems, limited commercial promotional efforts, dominance of existing marketing forces, high cost of integration and migration, perceived vulnerability due to the openness of source code, lack of OSS Policy / framework, lack of cost effective support services, lack of motivation, lack of capacity building & awareness and lack of awareness on Total Cost of Ownership (TCO).
Opportunities	**Threats**
Include low barrier to entry, economic opportunities for local industry, better reuse, better suitability, better support from OSS community, wider choices on OSS, ability to drive cross-industry collaboration and forging for new and better solution	Include resistance from the existing market forces, lack of awareness of OSS among decision-makers, inadequate support services, reduced activity of the community, incompatible versions, inadequate skilled-staff

The detailed SWOT Analysis of OSS Adoption along with ways to mitigate weaknesses and threats is given in <Annexure-III> “SWOT Analysis of OSS Adoption”.

3.6 OSS Current Scenario

1. International Scenario

Open competition from OSS support service providers bring a whole new dimension to the business models of OSS. As per Research Study in 2013 by Yeoman Technology Group and Linux Foundation[5], Linux usage in Mission-Critical applications has grown drastically to 73% in 2013, mainly due to growth in Cloud / Virtualisation and Big-Data... Netcraft's April 2013 Web Server Survey[6] shows the combined world market share of Apache and Nginx web-servers as 65%. As per Gartner Survey[7], August 2012, the market share of Android is 43%. In entertainment sector too, many movie / animation industries [8] like DreamWorks, Pixar, Weta Digital etc. uses OSS.

In a recent (Goldman Sachs, IDC) 2013-Survey[9] on common computing platforms (combined market for desktop, laptop, tablet and smart-phone), Linux has more than 40% market.

Gartner [10] report predicted that Google's Android-Linux platform would be installed on more than one billion device by 2014, giving increased dominance to Android; by 2017, shipments of Android devices would "dwarf" those of CSS based PCs and phones.

OSS presents significant opportunities for Government and many initiatives are being taken world-wide for OSS adoption. Led by UNDP and European Union, countries like USA, UK, South Africa, China, Brazil, Malaysia etc. are implementing nationwide policies or legislation promoting OSS. <Annexure-VI> "Adoption of OSS – International Scenario" outlines major initiatives on the adoption of OSS world-wide.

[5] Linux use in business, 2013 http://www.linuxfoundation.org/publications/linux-foundation/linux-adoption-trends-end-user-report-2013

[6] Web Server Survey, 2013 http://news.netcraft.com/archives/2013/04/02/april-2013-web-server-survey.html

[7] Gartner Report on Worldwide Sales of Mobile Phones, Aug. 2012, https://www.gartner.com/newsroom/id/2120015

[8] Linux in film production, https://en.wikibooks.org/wiki/Movie_Making_Manual/Linux_in_film_production#Filmmakers

[9] IDC, Goldman Sachs Research Report, Dec., 2012 http://seattletimes.com/html/microsoftpri0/2019853243_goldman_sachs_microsoft_os_has_gone_from_more_than.html

[10] Gartner Report on Smart-phones and Tablet rise, April, 2013, http://www.theguardian.com/technology/2013/apr/04/microsoft-smartphones-tablets

2. **Indian e-Governance Scenario**

OSS is adopted in many e-Governance projects executed by various Government Agencies in India. The details of initiatives from some of the public organisations like DeitY, State Governments, NIC and C-DAC are given in the <Annexure-VII> “Adoption of OSS – Indian e-Governance Scenario”.

3.7 Factors Influencing the Adoption of OSS in Government

The factors which influence the adoption of OSS in a positive manner are known as facilitators (indicated with “+”). On the other hand, the factors which negatively influence the adoption of OSS are considered as barriers or inhibitors (indicated with “-”).

The common influencing factors for adoption of OSS in Government Organisations are described below;

Classification of Influencing Factors

I. **Economic Level Factors**

(a) Cost Effectiveness (+)

(b) Preservation of Foreign Exchange (+)

(c) Enhanced Competition (+)

(d) Freedom to Use & Reuse (+)

(e) Help Innovation (+)

(f) Better Local Capacity Building (+)

(g) Minimised Piracy (+)

(h) Low Barrier to entry (+)

(i) Economic Opportunities for Local Industry (+)

(j) Better Reuse (+)

(k) Better Suitability (+)

(l) Wider choices on OSS (+)

II. **Security Level Factors**

(a) Enhancing Security (+)

(b) Enhancing source code level security without mistrust code (+)

III. **Technological Level Factors**

(a) Technological Compatibility based onStandards (+)

(b) Availability of Device Drivers for OSSOperating Systems (+)

(c) Relative Advantage of OSS (+)

(d) Trial ability of OSS (+)

(e) Availability of OSS stack (+)

(f) Technological Complexity in OSS usage (-)

(g) Presence of Proprietary Lock-in (-)

(h) Freedom to modify and improve (+)

IV. **Organisational Level Factors**

(a) Management's Positive Attitudes towards OSS (+)

(b) Champions of OSS (+)

(c) Size of Organisation (+)

(d) Diverse Expertise at Management Level (+)

(e) Inter-connectedness of Organisation (+)

(f) Organisational Slack on Resources (+)

(g) Inclination towards Business Processes Re-engineering (+)

(h) Availability of Internal Technical Expertise (+)

(i) Level of Formalisation (-)

(j) Centralisation on Decision Making (-)

(k) Availability of Financial Resources (-)

(l) Outsourcing impact (-)

V. **Environmental Level Factors**

(a) Rules for the adoption of OSS (+)

(b) Provision for Capacity Building (+)

(c) Availability of Support Services on OSS (+)

(d) Competitive Pressure (+)

(e) System Openness (+)

(f) Past Experience on OSS (+)

(g) Availability of Internal Collaboration Mechanism (+)

VI. **Individual Level Factors**

(a) Level of Organisational Objectives Consensus (+)

(b) User's Fear on De-skilling of Legacy Expertise (-)

The details of economic factors and security factors are made available in the “Annexure-III SWOT Analysis of OSS Adoption” and Section 10 “Security” respectively. Whereas, some of the influencing factors such as technology factors, organisational factors, environmental factors and individual factors are listed in the “Annexure-IV Common Influencing Factors for the Adoption of OSS”.

Need for Evaluation of Factors

The effects of each factor may vary from country to country; hence, the influence of each factor should be analysed for local environment. Factors having the greatest impact on the adoption of OSS are to be found and given highest priority.

The application context is also to be accounted for analysing the impact of each factor. The production systems is classified based on the strategic importance, into strategic, mission critical, routine-support and experimental / laboratory. The factors with their priority & inter-relationship are to be evaluated with reference to application context through appropriate methodology and metrics.

3.8 Impact of adoption of OSS in Government

Many Governments worldwide have started adopting innovative solutions offered by OSS in their e-Governance Systems. A recent survey analysis[11] says that about 35% of OSS adopters are Government agencies.

Reuse of ICT assets

Reuse of ICT Assets is easily facilitated by the adoption of OSS. For example, the use & reuse of OSS Stacks in applications, hosted at data-centres, without additional licensing costs, would bring down a huge amount of expenditure.

Reuse of ICT assets is being mandated by several Governments / their agencies worldwide. For example, UK Government[12], and US-DoD[13]. In the recent survey[14], it was estimated that the annual savings for European Union due to the adoption of OSS is about 450 billion Euro.

The details of benefits due to the adoption of OSS are given in <Annexure-III> “SWOT Analysis of OSS Adoption”.

Huge Employment Generation due to new ICT services

OSS solutions can generate very large employments in the ICT service areas. Small and Medium Enterprises (SME) and Public Sector Units (PSU) from India can be easily engaged in the ICT services based on OSS solutions. Many other groups directly or through the franchisee, with non-ICT backgrounds, can offer on-site services for managing the ERP training, data entry, reports, etc. The service-consumers will be in the order of several millions of citizens.

[11] Future of Open Source-2013, Survey Results, Black Bridge and Black Duck Software, 2013, http://www.blackducksoftware.com/news/releases/seventh-annual-future-open-source-survey-results-show-culture-quality-and-growth

[12] All about Open Source - An Introduction to OSS for Government IT, Version 2.0, April 2012, https://www.gov.uk/Government/uploads/system/uploads/attachment_data/file/78959/All_About_Open_Source_v2_0.pdf

[13] Open Technology Development (OTD) - Lessons Learned & Best Practices for Military Software, 16/05/2011, http://mil-oss.org/otd

[14] Contribution of open source to Europe's economy: Euro 450 billion, https://joinup.ec.europa.eu/news/contribution-open-source-europes-economy-450-billion-year

3.9 Types of OSS Support Models

Generally software support is required for operations and source code level modifications/enhancements. Engagement of vendor for OSS support is also follows similar approach as described below:

a) Operational support for software:

Operational support is a mechanism to run the software with the day-to-day operational requirements; also designing/developing applications based on the software but not involving modification or customisation to the source code.

b) Source code level support:

Source code level support is a mechanism to update or enhance the source code of the software to support additional features, to meet security requirement or fix vulnerabilities and bugs.

For most of the Government applications, operational level support is only required. For most OSS software, operational level support can be availed from multiple vendors within the country.

Government applications rarely require source code level support. Source code support is generally available from the communities/vendors for the respective Open Source Software. Availing the support from the community provides the advantage of staying with the original software distribution. Engaging third parties for source code level support may lead to branching from the software distribution of the mainline community/vendor. To maintain the branched version of the software requires additional effort, technical expertise and resources. Therefore, it is prudent to use community support as that is the practise worldwide.

Governments, in general, prefer to have more number of competing vendors to get quality support services on the chosen OSS as the multiple-support approach offers more flexibility and enhanced competition. The four most common types of OSS source code level support models are out lined below, along with their flexibilities. This helps to select the right support model for the OSS chosen by the Government[15]:

[15] http://www.openlogic.com/blog/bid/156899/Selecting-Your-Open-Source-Support-Vendors-And-What-Their-Business-Model-Means-to-You

1. **Pure Open Source**

   Selling of "Support and Services" is the main feature of this model. No vendor lock-in exists.

   Under this model, the OSS solution is managed / driven mostly by a community / foundation; for example, each of the OSS solutions like Apache-HTTP, PostgreSQL, Drupal, Eclipse is managed by a separate OSS community / foundation. A single edition of the OSS is released & maintained by the OSS Community. In general, there is no branding / marketing / certification of the OSS solution by the community. Multiple competing vendors offer support services / certification on each OSS solution. For example, PostgreSQL is supported by many professional support service companies[16]. The Government is not locked in to a single-vendor for availing the support services on the OSS. This approach gives the most flexibility for the Government since, they can decide at any time to avail the operational support services from any other vendors or through internal experts. Source code level support is community based.

2. **Certified Distribution Model**

   Selling of "Subscriptions, Solutions and Support" is the main feature of this model. Some level of vendor lock-in exists.

   The OSS solution is managed / driven mostly by a single-vendor company under this model. Example is the RHEL Operating System, which is managed / driven by Redhat. Several editions are released and maintained by the single-vendor company. The single-vendor company takes an OSS edition (for example, Fedora) and do additional testing, branding, certification or bundling, and the company releases the certified paid-for-fee edition (for example, RHEL) for the enterprises. There is no provision to have multiple competing vendors to offer support services / certification on the paid-for-fee edition. Only, the single-vendor company or its authorised franchises are allowed to offer support services. Hence the Government is locked in to a single-vendor company for availing the support services. This approach gives lesser flexibility to the Government.

[16] Professional Support on PostgreSQLhttp://www.postgresql.org/support/professional_support/

3. **Open Core Model**

Selling of “Subscriptions” for a Proprietary Version is the main feature of this model. Same level of vendor lock-in, as in CSS, exists in this model.

Under this model, the OSS solution is managed / driven mostly by a single-vendor company. Example is the “PostgresPlus Advanced Server” which is managed / driven by a single-vendor company, EnterpriseDB. The single-vendor company takes the OSS edition (for example, PostgreSQL Community Edition) as a core and creates a separate layer by adding additional functions, testing, branding, certification or bundling; and the single-vendor company releases the additional layer along with the core as a paid-for-fee for the enterprises. The source code of the value added layer is not released under OSS license. Only, the single-vendor company or its authorised franchises are allowed to offer support services on the paid-for-fee edition. Hence the Government is locked in to a single-vendor company. The Open Core model is similar to proprietary software model except that the core is released under OSS license. The use of such model in e-Gov is not generally preferred.

4. **Multiple Licensing Model**

In multi license model, the software is made available in two or more licenses with different terms and conditions. Usually the copyright owner of the software releases the software under copyright license which enables creating or deriving proprietary version of the software by copyright owner; while other licences would be based on copy left license which requires any derived work to be released under the same license. The complete control (including copy right to the source code, intellectual rights, trademarks, etc) of the OSS project is held with a single-vendor company. The single-vendor, in general does not allow the modifications to the source base. In case the single-vendor allows such modifications, the contributor has to transfer the copyright to the single-vendor.

Retaining Flexibility

The OSS based application will be given rating using a suitable rating mechanism based on the criticality of the application. The support model will be chosen based on the ratings.

Government needs to ensure continued-support for the open source solution even if a vendor decides to terminate support to it. If multiple vendors compete to offer support services to the open source solution, it is good for the Government since it increases the competition. On the other hand, if a single-vendor company controls the open source solution, then there is more risk of switching to alternative company in order to get the continued-support on the open source solution. Except the "Pure Open Source" model, all other support models, in general, are controlled by single-vendor and hence pose a potential risk.

The major motivation for adopting OSS is to have multiple choices for the software solutions and more competition but without any lock-in. If any support model creates the lock-in under the name of OSS, the major purpose for opting OSS is defeated.

3.10 OSS Licenses

Basics of OSS Licenses

This section suggests a simple and effective classification and management of OSS solutions based on the category of licenses. The classification terms[17] are commonly used from the point of adopters of OSS.

Based on the conditions / protections available on the OSS solution, the OSS licenses are classified [18] as Highly Liberal, Liberal, Less Protective, Protective and Highly Protective licenses with the restrictions increasing respectively. Legal advice to be sought is also based this level. Liberal type license is also known as Permissive, Non-Viral or Academic license. Protective license type is also known as Reciprocal, Restrictive, or Copy Left license.

All types of protective licenses (like GPL, LGPL, and AGPL) ensure the availability of modified OSS libraries under OSS license. The Liberal licenses (like Public Domain, MIT, BSD, Apache) restrain the release the modified OSS libraries under OSS license.

[17] Free And Open Source Software Licensing Primer, by "Shun-ling Chen", Published by IOSN & UNDP-APDIP and Elsevier, 2006, ISBN-13: 978-81-312-0422-1 ; ISBN-10: 81-312-0422-7, http://www.iosn.net/licensing/foss-licensing-primer/foss-licensing-final.pdf

[18] OSS Licensing Overview, http://opensourceforamerica.org/learn-more/oss-licensing-overview/ ; The Mozilla Public License Version 2.0: A Good Middle Ground?, http://julien.ponge.org/notes/mozilla-public-license-v2-a-good-middleground/

The OSS with liberal licenses can be used along with other applications / libraries which have OSS licenses or CSS licenses.

Protective licenses (like GPL-v2) allow users to run, copy and modify the software, and distribute the modified software. However, users are not allowed to add their own restrictions. Also the modified software must be released under the same licensing terms.

Less Protective (like LGPL, MPL, EPL) license allows linking an unmodified OSS library to any application / library. Hence the use of unmodified OSS library (with licenses like LGPL, MPL, EPL) does not require the release of the application source to be open. The license obligations of OSS are to be adhered and necessary legal opinion may be sought.

Some public agencies, especially in USA[19] and European Union[20], prefer to publish all software developed for any government department, under OSS licence. This model of releasing e-Governance application under OSS license allows the use of all types of OSS licenses (including Protective licenses like GPL / AGPL).

Why we need it?

- To implement the objectives of the Policy on Adoption of Open Source Software
- To implement the objective of the National Policy on Information Technology, 2012
- To widen the adoption of OSS to cover various projects
- To improve the ecosystem of OSS
- To minimise the informal use and absorption of OSS
- To mitigate the risks like hidden lock-ins and poor maintainability

[19] Open Technology Development (OTD): Lessons Learned & Best Practices for Military Software. 2011-05-16 http://dodcio.defense.gov/Portals/0/Documents/FOSS/OTD-lessons-learned-military-signed.pdf

[20] Introduction to the EUPL licence https://joinup.ec.europa.eu/software/page/eupl/introduction-eupl-licence

Overview of OSS Licenses

The commonly used OSS licenses are depicted in the following matrix below. In this, the rows indicate different types of licenses and columns indicate different usage

Environment for OSS-library Use ➡ License Type ⬇	OSS-library hosted without modified source	OSS-library hosted with modified source	OSS-library distributed to customer without modified source	OSS-library distributed to customer with modified source
Highly Liberal (Public Domain, MIT)	🟪	🟪	🟪	🟪
Liberal (Apache-v2, BSD(New))	🟪	🟪	🟪	🟨
Less Protective (LGPLv2, MPL, EPL, LGPLv3)	🟪	🟪	🟪	🟦
Protective (GPLv2, GPLv3)	🟪	🟪	🟦	🟦
Highly Protective (GPL3 Affero)	🟦	🟦	🟦	🟦

Violet denotes the OSS license with less or no restrictions for the particular environment.	🟪
Yellow denotes the OSS license with moderate protection for the particular environment.	🟨
Blue denotes the OSS license with more protection for the particular environment.	🟦

The detailed description of these licenses and guideline for selecting the appropriate OSS libraries based OSS licence type can be referred at UNDP Report on OSS Licensing[21]

[21] FOSS Licensing http://www.iosn.net/licensing/foss-licensing-primer/foss-licensing-final.pdf

How will it be implementation?

- Creation of Institutional Mechanism
- Partnership with industry (including SMEs)
- Partnership with OSS communities in India & abroad
- Engaging academia
- Collaborative mechanism
- Provisioning of support services on OSS
- Target groups for services on One Stop Solutions on approved OSS stacks

3.11 Interoperability & Open Standards

Open Standards plays an important role in fostering healthy competition, enhancing the interoperability among e-Governance Systems and better communication among all stakeholders.

Open Standard is defined by each country or Public agency. Government of India has brought out "Policy on Open Standards for e-Governance" in November, 2010 to enhance the standardisation activities in India[22].

i. OSS and Open Standard

"Open Standard", in general, refers to a technical specification as a result of consensus during formulation and ratification stages.

OSS refers to the implementation of technical specification by a community using Open Source licensing and collaborative contributing model;. The licensing and contributing model may vary from one community to another.

Though OSS and Open Standard concepts are similar in terms of availability of specification, cooperative development-model but still there are some differences.

ii. Significance of Open Standards on OSS

Migration from CSS to OSS and vice-versa is made easier by Open Standard. Mandating Open Standards has a complementary effect on OSS systems, introduces increased competition and facilitates better compatibility between CSS & OSS.

The availability of an OSS reference implementation would spur quicker

[22] Policy on Open Standards for e-Governance, https://egovstandards.gov.in/policy

adoption and acceptance of the standards as the implementation of the standard is available for reuse. Examples include HTML5, JavaScript, etc.

3.12 Security Aspects in OSS

1. Multi-User based OSS System

OSS systems are mostly based on the multi-user, network-ready UNIX model which has a strong security and permission structure. Even then vulnerabilities in applications result in limited security breach in OSS systems. But, availability of the source code for OSS systems helps the developers to discover and fix vulnerabilities.[23]For example, TCP/IP, HTTP, DNS, SMTP & IMAP.

2. Vulnerability & Bug-Fixes

Since bugs and security vulnerabilities are disclosed in OSS the service-providers can fix bugs and vulnerabilities in OSS source code. Whereas in CSS systems the CSS vendors are involved in bug fixing activities. In general, well-known OSS has potential for faster release-cycle of bug-fixes and the security of OSS is better because the bug and security vulnerabilities are frequently fixed within the respective OSS Community. The security practices are often backed by Commercial support services agencies that also support indemnification; this has a dramatic effect on the roll-out of the systems which are based on OSS.

3. No Hidden Malicious Code

The security-threats, like hidden back-doors or holes in software, in current ICT infrastructure have encouraged many Government organisations to switch over to OSS. Intentional hiding of security-holes is very rare in OSS and is detectable due to review process. Thus by minimizing security threats, strategic control is far better with the use of OSS.

4. Establishing Enterprise Security with OSS

OSS Systems tend to be generally more secure and are being used by banks, finance and insurance companies[24].

Organisations [25] need to ensure that the right level of expertise exists with

[23] Why FOSS? - http://en.wikibooks.org/wiki/FOSS_A_General_Introduction/Why_FOSS%3F

[24] Wall Street Opens Doors to Open Source Technologies - http://www.wallstreetandtech.com/it-infrastructure/wall-street-opens-doors-to-open-source-t/217400216

[25] Section 4.1, A Guide to Open Source Software for Australian Government Agencies,

all types of support providers including in-house experts. Adequate maintenance and support services should be made available for OSS as in the case of CSS, in order to minimise the risk.

A central core group of ethical hackers should continuously look into the vulnerabilities and loop holes of OSS solutions. Support should be taken from communities and Non-Profit Organizations promoting OSS who provide security patches/updates.

The OSS solutions should be tested for security threats by Academic community of Computer Science and the issues if any may be used for student projects to get the solutions.

Other security implications exists both in OSS as well as CSS, like older versions getting outdated and no longer having support from respective communities. Some of the generic security guidelines are as follows:

(a) Protect network with a strong firewall

(b) Secured Remote Access

(c) Securing Data on local desktops, laptops and tablets using encryption

(d) Securing Wi-Fi access points

(e) Adopting Best Practice for System Administration

(f) Secured Internet Access from Intranet through Web Proxy.

The above guidelines are described in <Annexure-V> “Guidelines for Establishing Enterprise security with OSS”.

3.13 Unified Software Development for Mobile, Tablet & Desktop

Traditionally, e-Governance applications have been developed for desktops and then customised for various types of mobiles & tablets using native approach.

1. Mobile-Native Approach

The native traditional applications were opted in the early years for mobiles; this created native applications for specific mobile platform using its native Software Development Kits (SDKs) & languages. One has to learn different OS, their SDKs & programming-languages if the application is expected to run on different types of mobiles like Android, Apple, Symbian, Window Phone, Blackberry, etc. This approach utilises the native features of the mobiles effectively.

Version 2.0, June 2011 - http://www.finance.gov.au/files/2012/04/AGuidetoOpenSourceSoftware.pdf

2. Emergence of Alternative Approaches

The explosion of varieties and types of mobiles, especially smart-phones with HTML5 browser, has challenged native applications adoption. In 2011, there were about 336 million HTML5 capable mobiles sold. As per the report [26] , Research firm Strategy Analytics forecasts that one billion HTML5 capable mobile devices would be sold in 2013. ABI Research sees more than 2.1 billion mobile devices with HTML5 browsers by 2016. IDC estimates indicate that over 80 percent of all mobile applications would be wholly or partly based on HTML5 by 2015.

Alternative approaches are being explored to simplify the application development process since there has drastically changed due to the emergence of HTML5 based Open Web Technology (OWT) and Cloud Technology.

OWT characteristics are as follows:

- Adherence to Web Standards,
- Wide-adaptability,
- Develop & run the same on all devices,
- Provision of separation of presentation and logic,
- Facility to create rich client with highly scalable thin server,
- In-built methods based on standards to send software updates,
- Provision to exploit the generic and native features of mobiles.

3.14 Rapid Application Development Environment for OSS

The manually edited software is highly efficient for building, maintaining and modernising business-critical Web 2.0 applications. However, it is difficult to follow the same process every time as it takes more time to deliver the solution. A Rapid Application Development (RAD) environment with visual, WYSIWYG development studio or a set of reusable drag-and-drop components / templates is required to meet quick delivery schedule.

In general, RAD solutions are used for the development of OSS applications to meet quick delivery schedule.

3.15 Localisation and OSS

Localisation involves taking a software product and making it linguistically and culturally appropriate for the target

[26] What to Expect from HTML5 in 2013, by Fahmida Y. Rashid, December 9, 2012, http://html5center.sourceforge.net/blog/What-to-Expect-from-HTML5-in-2013

country/region where it would be used and distributed. OSS has an advantage in this area because of its open nature. Users are able to modify OSS to meet the localisation requirements of a particular region.

Localised version of any OSS helps in reaching out to the rural population and the people living in remote areas in India, thus bridging the digital divide in the country.

C-DAC has indigenously developed, Bharat Operating System Solution (BOSS), an OSS based OS with Indian language interface. Bharateeya Open Office developed by CDAC supports Indian languages. Industry in India is also aggressively working on localisation efforts. Major South eastern Asian countries like China, Japan and Korea are also actively pursuing OSS localisation.

3.16 Device Driver

When implementing e-Governance systems, the Device Drivers are available for Windows Operating Systems (OS) as a default. However, Device Drivers are not easily available for GNU / Linux Operating Systems which is also widely-used in Computers and Peripherals deployed in the roll out of e-Governance systems. Users should ensure the availability of device drivers for GNU Linux Operating systems while procuring Computers and associated Peripherals.

3.17 Procurement Guidelines

Standardised common methodology should be developed for rating OSS against another OSS for Indian scenario as indicated on <Annexure-VIII> “Rating of OSS against other CSS using Business Metrics". A set of guidelines on inclusion of clause related to OSS solutions in procurement should be brought out.

1. Guidelines for Procurement

Some of the important factors, which could be considered for the inclusions in tender terms and conditions while procuring / selecting ICT solutions, are given below: Preferred Option - OSS should be considered as a preferred option.

(a) **Essential functionalities** – To save resources only the required functionalities should be specified, instead of over-specifying the requirements.

(b) **Customisation Cost** - If the solutions to be acquired need further customisation for adoption, then the factors like cost of customisation, support & maintenance cost, flexibility on engaging competing agencies, legal / licensing obligations, etc. should also be considered.

(c) **Security** - The security requirements of the solutions should be consivdered on a case-to-case basis.

(d) **Survival-ability** – The planned continuity of the solutions with further developments till their life-cycle mitigates the risks related to change over to another solution in future.

(e) **Compliance with Open Standards** – The compliance on Open Standards should be mandated

(f) **Transferability / Reuse** – The flexibility of using / reusing the solution in different scenarios (use in conventional systems, virtual machines, cloud systems, emulated systems, etc.), locations (anywhere in 3-tier Government Architecture) and its financial implications should be obtained.. Appropriate structure and guidelines need to be established for shared solutions on e-Governance application between Government / Public agencies through efforts like e-Gov-AppStore, Mobile-Seva-AppStore.

(g) **Maturity** - Its adaptability, activity, longevity, services available on it, documentation, integration, security, skill set availability should be considered.

(h) **Maintenance and support services** - The quality level of support and maintenance services expected to meet the requirements should be specified in the tender specifications as a mandatory condition to mitigate the risks.

(i) **Lower barriers for SME** - Appropriate steps should be taken to avoid the elimination of firms with good OSS skills and track records from tendering processes based on turn-over conditions. Separate tender conditions (like years of operations, turn-over and number of manpower, number of projects executed) should be set with appropriate lower values for encouraging the participation of SMEs.

The relevant factors are required to be analysed and documented for procuring / selecting ICT solution.

2. Rating of OSS

If the OSS solution is to be evaluated against CSS solution, then models like (i) Total Cost of Ownership (TCO), (ii) Return on Investment (RoI), (iii) Internal Rate of Return (IRR) could be considered. If required, these models could be analysed to select / customise a suitable model; these are discussed in <Annexure-VIII> “Rating of OSS against other CSS using Business Metrics".

The selection process for selecting a suitable OSS is discussed in <Annexure-IX> “Rating of OSS based on Performance matrix".

3. Total Cost of Ownership

In general, only the software licensing cost is considered while acquiring CSS or OSS. However, other costs towards search, exit, transition, additional hardware, training etc., are also to be accounted under the Total Cost of Ownership (TCO)[27]; this gives the overall picture of the savings resulting from the use of OSS. Cost comparison model should address factors like investing money in local IT industry for availing support services instead of acquiring software, enhanced local ecosystem (SMEs, Knowledge base), preservation of foreign exchange, improved negotiating power of entire Government as a single entity, etc. All assumptions should be specified while calculating the TCO. The metrics along with other technical points influence the decision-makers to opt for OSS solution while developing e-Governance systems. The details of TCO are given in <Annexure-VIII> “Rating of OSS against CSS using Business Metrics". Suitable TCO model, after customization to suit local conditions, should be brought out.

3.18 Stages for induction of OSS Solution

Stages for the induction of OSS solution include the following;

(a) **Exploration & Testing:** First of all the available set of OSS solutions need to be explored. The required ones may be filtered based on some key parameters such as type of license, functionality, availability, longevity etc., The filtered OSS software solution may be downloaded and installed to make it work as per the instructions given in the documentation. Then it needs to be tested for its functionality, performance, security etc. Finally the tested solutions meeting the benchmarks may be selected for PoC.

(b) **Proof of Concept (PoC) for confidence building:** For confidence building the facilities and

[27] Total Cost of Ownership of Open Source software: a report for the UK Cabinet Office submitted by Shaikh, Maha and Cornford, Tony, London School of Economics and Political Science, 2011, http://eprints.lse.ac.uk/39826/

functionalities of the selected OSS solution are required to be shown in some of the Projects. Thus it is required that PoCs are conducted to explore capabilities of these solutions for various project requirements.

(c) **Training & Hand-holding:** Once the OSS solution is made ready for a project, training should be given to the concerned project teams, so that further development and maintenance becomes easier. User manuals, Technical Documents should be prepared and handed over to the project team. Backup mechanisms, recovery mechanisms should be mentioned clearly.

(d) **Roll-out in live Systems:** While implementing the tested solution in LIVE systems, proper and routine monitoring should be done. Regular backup of application-data should be taken. The OSS solution should be maintained in the repository.

(e) **Creating Multiplication Effect:** The OSS solution once implemented in one project should be reused for other similar projects with some customisations as per the project requirement.

3.19 Proposed Ecosystem for Promotion of OSS

Ecosystem includes Institutional Mechanism, Partnership with Industry, Academia and OSS Community. Support services would be provisioned and collaborative mechanism solutions will be established.

i. Creation of Institutional Mechanism

(a) Apex Body should drive the OSS initiatives; the stake-holders include DeitY, NIC, CDAC, STQC, Industry representatives, nominated officials from line Ministries of Centre, State Governments and R&D Institutes. Academia and OSS Communities should be linked suitably. The uniform guidelines should be prepared in the consultative mode and it should be adopted by all stakeholders to eliminate duplicate efforts. This would facilitate better interoperability / integration of e-Governance systems.

(b) The entire program may be sub-divided into few sub-programs and each sub-program may be executed by separate public agencies such that they complement each other. Necessary funds, human-resources and hired-resources should be provided to offer adequate support services, consultancy services on the adoption of OSS through help desk.

(c) Expert Committees / Specialist Committees should be formed under program implementing agency and they shall be assigned the tasks related to OSS Stack, etc. The Committees would submit the draft reports for obtaining feedback from stakeholders. They would update the drafts and submit to the Apex Body for ratification.

(d) Key Stakeholders for sustaining the momentum on OSS Adoption would comprise of Senior Management, Project Managers, System-Developer, System-Integrators, Service-Providers, Product-Partners, Technology Experts, End-Users and Consultants; these are outlined in <Annexure-X> “Key Stakeholders of Ecosystem"

The awareness programs on OSS adoption in e-Governance Systems would be offered to the experts from the Ecosystem. Detailed capacity building programs would be offered to System-Developer, System-Integrators, Service-Provider and Technology Experts from Government organisations.

ii. Partnership with Industry (including SMEs)

A forum may be created for the collaboration between Industry (including SMEs) and Government users in order to have better understanding on requirements and capabilities in adopting OSS. Some of the expected services from Industry are;

(a) Development, Staging and Maintenance of e-Governance applications using OSS Stack

(b) Publishing information, maintaining knowledge repository & creation of awareness about OSS

(c) Capacity Building on OSS

(d) Maintaining repository for each component of OSS Stack

(e) Creation and Offer of pre-configured, integrated and packaged OSS Stack for use & reuse at data centres

(f) Supply of hardware with pre-installed OSS operating system & solutions

(g) Development of particular OSS solution to fill the gaps, if needed.

(h) Support on achieving strategic objectives of government rather than direct cost benefit

iii. Partnership with OSS Communities in India & Abroad

Government may consider sponsoring the activities of OSS Community. The type of sponsorship may be in any of the forms listed below:

(a) Creating Repository/Mirror sites of OSS solutions listed in the OSS Stack

(b) Providing hosting services

(c) Providing Human Resources / Code/ Documentation contributions

(d) Subscribing membership

(e) Sponsorship for the travel of experts from abroad to participate in conferences/workshops/trainings/seminars in India

iv. Engaging Academia

Sponsorships for Student Projects used in e-Governance (Development/ enhancement of OSS solutions/products/Documents).

(a) Incentives for faculty for managing OSS projects

(b) Awards for best Open Source Student Project

(c) Award for Institute – Contribution to OSS

(d) Awareness / Capacity Building Program on OSS

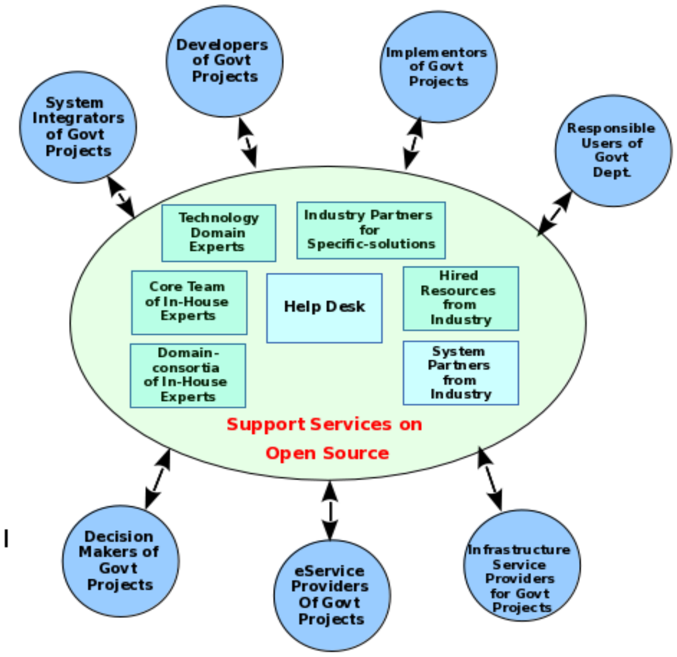
It is proposed to form Working Groups to enhance OSS course development, e-learning and collaborative learning, application of Open Source methodology and business models for real world scenarios in e-Governance.

The courses will include, philosophy & methodology in OSS, software engineering based on OSS, use of OSS Desktop applications and Linux OS, OSS Servers (including servers for Web, Application, Database, Infrastructure) & OSS Applications based on them, Software Development Solutions; the courses may be at the certificate level, degree level and post-graduate level.

The community approach used by some Indian institutes [28] can be considered for the generation of trained manpower.

The working groups should include OSS Technology Experts, Teachers and Academicians.

[28] Spoken Tutorial, Indian Institute of Technology, Bombay, Mumbai at http://spoken-tutorial.org


Developers of Govt Projects
Implementors of Govt Projects
System Integrators of Govt Projects
Responsible Users of Govt Dept.
Technology Domain Experts
Industry Partners for Specific-solutions
Core Team of In-House Experts
Help Desk
Hired Resources from Industry
Domain-consortia of In-House Experts
System Partners from Industry
Support Services on Open Source
Decision Makers of Govt Projects
eService Providers Of Govt Projects
Infrastructure Service Providers for Govt Projects

v. Collaborative Mechanism

Enhanced Collaborative mechanism (like help desk, knowledge portal, issue tracking system, discussion forums, e-mail support, and telephone) should be established for the adoption of OSS. Preparation of reports, creation of central repository of components of OSS Stack and integration methodologies should be carried out with the support of Industry & Academia for sharing with other stack holders.

vi. Provisioning of Support Services on OSS

The proposed division should provide multi-level support for the adoption of OSS as listed below:

(a) Help-desk,

(b) Core-team and domain-consortia as part of in-house experts,

(c) Hired-resources from Industry,

(d) System-Partners from Industry (who run the operations),

(e) Specific-Solution-Partners from Industry (who fix/enhance the source code of the OSS) and

(f) Technology Domain experts from Community, Academia, R&D Institutes and Government.

In addition to the central mechanism for support services, the Government should take initiatives for setting up OSS Support centres throughout the country. Services from Industry should also be utilised for this purpose.

In-House Experts should work on exploration of technology, internal support and domain-consortia forums.

vii. Target Groups for Services on One Stop Solutions on Approved OSS Stacks

The Services can be availed by

(a) System Integrators of Government Projects

(b) Developers of Government Projects

(c) Implementers of Government Projects

(d) Responsible Users of Government Department

(e) Decision Makers of Government Projects

(f) e-Service Providers Of Government Projects.

(g) Infrastructure Service Providers for Government Projects.

viii. Promotional Mechanism on the Adoption of OSS

(a) Provisioning of bundled & identified OSS Stack with appropriate fine tuning, hardening and security patches. The stack can be reused in software development, staging and deployment environments on virtual images / clouds available in other locations. The stack should also be provided with support services and source-code level enhancements. This will motivate the e-Governance implementers to come forward for the adoption of OSS.

(b) Capacity Building for in-house experts and policy makers by way of on-the-job training, class-room training programs and work-shops should be conducted.

3.20 Proposed Ecosystem for Promotion of OSS

This section summarises the recommendations for the adoption of OSS.

i. Recommendations for Implementing Agencies for OSS Framework

(a) Preference should be given to select OSS libraries which have liberal and less restrictive license model.

(b) Selecting appropriate OSS stack for development of applications and infrastructure is crucial for performance and sustained support.

(c) Establish Multi-Level Support Services on the adoption of OSS.

(d) Provisioning of application development, staging and deployment environments for the reuse of Open Source Stacks with support services.

(e) Offer services for preferred areas and provide support.

(f) Continue R&D efforts in OSS in identified thrust areas.

(g) National repositories/ knowledge banks should be created for OSS solutions, technologies and applications.

(h) Development of two tool-kits (one tool-kit for rating OSS against another OSS and another tool-kit for rating OSS against CSS) should be brought out.

(i) Develop a mechanism/tool to rate the OSS based application based on the criticality of the application.

(j) Transferability of ICT Assets (which facilitate the reuse) with in all levels of Government and public agencies without additional expenses should be considered while procuring them.

(k) The distribution of the modified source code and executable of the OSS across various units of the single Government entity should be considered as internal distribution.

(l) Use of OSS in Government Departments along within skill development programs should be encouraged.

(m) The security of OSS solutions under OSS Stacks should be enhanced by creating a two layered internal & external audit mechanism and retrofitting mechanism under the proposed structure.

(n) OSS application development with Indian languages interface should be encouraged.

(o) Simpler & easier Software Development with GUI, Meta-Language and Templates should be provided, as a RAD environment, to achieve faster adoption of OSS in order to meet the quick delivery schedule.

(p) The guideline on influencing factors for the adoption of OSS should be brought out by customising for Indian Scenario.

(q) Enforcement guidelines on Open Standards Policy of Government of India should be brought out to accelerate the adoption of OSS.

(r) The model used by some Indian Institutes may be considered for creating training and learning materials using the community approach.

(s) Development of a community engagement model to encourage internal developers to participate in the open source community under the appropriate policies and engage with external developers

ii. Recommendation for E-Governance Project Implementation Teams

(a) Since many social, economic and strategic benefits are provided by the adoption of OSS, the OSS options should be considered seriously by the e-Gov planners, architects and developers.

(b) This Framework should be used to expedite the adoption of OSS in e-Governance in India.

(c) Focus on Preferred areas for adoption.

(d) Since many socio, economic and strategic benefits are provided by the adoption of OSS, OSS should be considered as a preferred option.

(e) Preference should be given to "Pure Open Source Model" for availing the support service on OSS.

(f) Government Agencies and Departments should seek to avoid vendor lock-in to proprietary IT products and services. RFP (Request for Proposal) documents should avoid using vendor specific product/brand names.

(g) Applications developed by the Government of India should be cross platform and not be locked in to a specific platform.

(h) For Government funded software research and developments in India, scientists/ researchers should be encouraged to publish their innovations under Open Source and Open Document licenses, except for security reasons.

(i) Large Projects should be split into smaller Projects for development by different parties/vendors/SMEs and integrated & implemented by the project teams. This will reduce the amount of resources required for the smaller project, encourage SMEs participation, reduce the risks in ICT projects and facilitate the adoption of OSS.

(j) Open Web Technology should be preferred to develop once and run the same on all devices. Device Specific Development (Desktop, Tablet, Mobile, etc.) should be discouraged.

(k) Code contribution to OSS community should be encouraged.

iii. Recommendations related to RFP/Procurement

(a) OSS Solutions should be considered as preferred option in IT procurements by Government of India. In cases where the merits of OSS and CSS are comparable, contracts could be awarded to OSS solutions in recognition of issues like value for money as well as enhanced strategic control, security, reuse, cost saving, knowledge society creation, adherence to Open Standards etc. which are hard to quantify.

(b) Vendors must provide justification for exclusion of OSS in their responses to RFPs (Request for Proposals).

(c) Hardware and peripherals procured by Government Agencies and Departments should have support for Open Source device drivers for ensuring interoperability of systems.

3.21 Annexure-I Typical OSS Stacks for Java, PHP and Open Web Technologies

This section lists the recommended Open Source Software Stack for developing and deploying e-Governance Applications. It also includes Open Web Technology (OWT) Stack for development of new projects to work on desktops, varieties of mobiles & tablets.

Legends

"xxxxxxx"	This notation indicates that the solution/language "xxxxxxx" is a well-accepted "core product".
🟩	This colour denotes set of Minimal Core OSS solutions for Application Development & Deployment
🟪	This colour denotes set of Minimal Core OSS solutions for Application Development Specific case & for Infrastructure
🟨	This colour denotes set of Additional OSS solutions for Building Mobile Native (OS-Android, iOS, Windows Phone, BlackBerry, Symbian) Applications (Development & Deployment) using HTML, CSS, JavaScript.

Note:

1. The software stack given below is updated in February 2015.

2. "No Discrimination" indicates that the set of tools under this column may be considered as the next best option after the tools in the column marked "Preferred".

			New Projects		Legacy Projects			
		Functional Areas for Tools	OWT Technology Stack		Java Technology Stack		PHP Technology Stack	
			Preferred	Remarks – No discrimination	Preferred	Remarks – No discrimination	Preferred	Remarks – No discrimination
Minimal Core Solutions	Solutions for Application Development & Deployment	Programming Language Client-side	HTML (5.0), CSS (3.0), JavaScript (1.8.x), Jquery (2.1.x)	HTML (5.0), CSS (3.0), JavaScript (1.8.x), Jquery (2.1.x)	HTML (5.0/4.01), CSS (3.0/2.1), JavaScript (1.8.x), Jquery (2.1.x)			
		Relational Database	PostgreSQL Community Edition^{C} (9.4.x)		PostgreSQL Community Edition^{C} (9.4.x/8.4)	MariaDB Community Edition (10.0.x)/ MySQL Community Edition^{C} (5.6.x)	PostgreSQL Community Edition^{C} (9.4.x/8.4)	MariaDB Community Edition (10.0.x) / MySQL Community Edition^{C} (5.6.x)
		Web Service Framework	Apache CXF (3.0.x) with Apache Tomcat^{C} (7.0.x)	Symfony (2.6.x)	Apache CXF (3.0.x) with Apache Tomcat^{C} (7.0.x)		Symfony (2.6.x)	CakePHP (2.6.x)
		Web / HTTP Server	Apache HTTP Server^{C} (2.4.x)	Nginx (1.6.x)	Apache HTTP Server (2.4.x/2.2.X)	Nginx (1.6.x)	Apache HTTP Server (2.4.x/2.2.X)	Nginx (1.6.x)
		Programming Language Server-side and Library	Core Java, OpenJDK^{C} (1.7)	PHP (5.6.x/5.5.x/5.4.x/5.3.x)	Core Java, OpenJDK^{C} (1.7/1.6)		PHP (5.6.x/5.5.x/5.4.x/5.3.x)	PHP (5.6.x/5.5.x/5.4.x/5.3.x)

		Server Side Framework			Apache Wicket Framework (6.x /1.5/1.4) with extensions	Struts (2.3.x) / Spring (4.x)	Symfony (2.6.x) with extensions	- CakePHP (2.6.x)
		Application Server					Apache HTTP Server (2.4.x/2.2.X)	Apache HTTP Server (2.4.x/2.2.X)
	Solutions for only Application Development	IDE	Eclipse^{C} (4.x) with extensions	- Netbeans (8.x)	Eclipse^{C} (4.4.x) with extensions	- Netbeans (8.x)	Eclipse^{C} (4.4.x) with extensions	- Netbeans (8.x)
		Source Code Control	Apache Subversion^{C} (1.8.x)	- Git (2.3.x)	Apache Subversion^{C} (1.8.x)	- Git (2.3.x)	Apache Subversion^{C} (1.8.x)	- Git (2.3.x)
		Documentation	LibreOffice^{C} (4.x)	- Openoffice (4.x)	LibreOffice^{C} (4.x)	- Openoffice (4.x)	LibreOffice^{C} (4.x)	- Openoffice (4.x)
	Solutions for Infrastructure	Server Operating System	CentOS^{C} (7.x)	Ubuntu(14.04/12.04/)	CentOS^{C} (7.x/6.x/5.x)	Ubuntu(14.04/12.04/)	CentOS^{C} (7.x/6.x/5.x)	Ubuntu (14.04/12.04)
		Desktop Operating System	Ubuntu (14.04)	BOSS (5.0) / Fedora (21.x)	Ubuntu (14.04/12.04)	BOSS (5.0) / Fedora (21.x)	Ubuntu (14.04/12.04.	BOSS (5.0) / Fedora (21.x)
		Authentication with Single Sign On	Central Authentication Service (CAS) (4.x/3.5.x)					
		Directory Services	OpenLDAP^{C} (2.4.x)					

	Solutions for Independent usage	Portal/CMS	Drupal^{C} (7.3.x)		Liferay Community Edition (6.x)		Drupal^{C} (7.3.x)	Joomla (3.3.x/2.x)
		Digital Archival Repository	Dspace^{C} (5.x)					
		Integrated Library Systems	Koha^{C} (3.18)					
		E-learning	Moodle^{C} (2.8.x)					
Additional Solutions	Solutions for Application Development & Deployment	Database Replication	SymmetricDS (1.7.16)					
		Building Mobile Native (OS- Android, iOS, Windows Phone, BlackBerry, Symbian)	Apache-Cordova (4.2.x) (PhoneGap)	Apache-Cordova (4.2.x) (PhoneGap)	- Not Applicable	- Not Applicable	- Not Applicable	- Not Applicable
		Build Tool	Apache Maven (3.2.x)		Apache Maven (3.2.x)		Phing (2.10.x)	
		GIS Server	Geo Server (2.6.x)		Geo Server (2.6.x)		Map server (6.4.x)	
		GIS Desktop	Quantum GIS (2.x)	GRASS GIS (7..x), gvSIG (2.x)	GvSIG (2.x)	Quantum GIS (2.x), GRASS GIS (6.4.x)	Quantum GIS (2.x)	GRASS GIS (7.x), gvSIG (2.x)
		GIS Database	PostGIS (2.x)		PostGIS (2.x)		PostGIS (2.x)	

		Non-Relational Database	Apache Cassandra (2.x)	- Apache Hbase (0.984) with Hadoop (2.6.0) - Apache CouchDB (1.6.x) (JSON Data Type only)				
		Object Relational Mapping	Hibernate (4.3.x)		Hibernate (4.3.x)		Doctrine (2.4.x)	
		Database Administration	PgAdmin III (1.20.x)		PgAdmin III (1.20.x)	PhpMyAdmin (4.3.x)	PgAdmin III (1.20.x)/php pgAdmin(5.1.x)	PhpMy Admin (4.3.x)
		Database Reporting	Jasper Report (5.6.x) with iReport Designer (5.5.x)	Birt (4.4.x)	Jasper Report (5.6.x) with iReport Designer (5.5.x)	Birt (4.4.x)	MPDF (5.7.x)	
	Solutions for Infrastruce	Virtualisation	Xen Server (6.5.x) / Xen Cloud Platform (XCP) (1.6/1.1)					

Cloud Platform

CloudStack (4.4.x)	OpenStack (Version 2014.2-Juno)	CloudStack (4.4.x)	OpenStack (Version 2014.2-Juno)				
	Video Conference	Apache OpenMeetings (3.0)		Apache OpenMeetings (3.0)		Apache OpenMeetings (3.0)	

Solutions for Application Testing	Testing	QUnit (1.17.x) JUNit (4.x) Apache Jmeter (2.12) W3C markup Validators service		QUnit (1.17.x) JUNit (4.x) Apache Jmeter (2.12) W3C markup Validators service		Phpunit (4.x) Apache Jmeter (2.12) W3C markup Validators service	

3.22 Annexure-II Illustrative list of OSS

	Functional Area for Solutions	OSS	
		Preferred	Optional
	Java Programming Language Environment	IcedTea	
	PHP Programming Language	PHP	
	Document type for simple Hyper Text Web Content	HTML 5	HTML 4
	Document type for complex Hyper Text Web Content	HTML 5	XHTML 1.1
	Cascading Style sheet	CSS 3	CSS 2
	Client Side Scripting Library	jQuery	
	Java Framework	Apache Wicket	Struts, Spring
	PHP Framework	Symfony	CakePHP
	Python Framework	Django	
	Java Application Server	Apache-Tomcat	Jetty
	Java Enterprise Application Server	Apache-TomEE	jBoss (Community Edition)
	Web (HTTP) Server	Apache-HTTP	Nginx
	PHP Application Server	Apache-HTTP with mod-php	
	RDBMS Database Server	PostgreSQL	MariaDB
	IDE for Java	Eclipse-JDT	NetBeans
	IDE for PHP	Eclipse-PDT	NetBeans
	Documentation	LibreOffice	Openoffice
	Source Code Control	Apache Subversion	Git
	Performance Load Testing	Apache Jmeter	
	Java Unit Testing	Junit	
	PHP Unit Testing	Phpunit	
	PHP CMS	Drupal	Wordpress, Joomla
	Java Object Relational Mapping	Hibernate	MyBatis
	PHP Object Relational Mapping	Doctrine	Propel

	RDBMS Database Administration	PgAdmin	PhpPgAdmin
	Virtualisation	Xen Cloud Platform	KVM
	Cloud Platform	CloudStack, Meghdoot	OpenStack
	Server Operating System	CentOS , BOSS Advanced Server , Debian	Ubuntu Server
	Desktop OS	BOSS, Ubuntu , Debian, Fedora	
	Authentication with Single Sign On	Central Authentication Service (CAS)	Shibboleth
	Digital Archival Repository	Dspace	
	RDBMS Database Replication	SymmetricDS	
	Java GIS Server	GeoServer	
	PHP GIS Server	UMN MapServer	
	GIS Desktop	Quantum GIS	GRASS GIS, gvSIG
	Java Build Tool	Apache Maven	Apache Ant
	PHP Build Tool	Phing	
	Integrated Library Systems	Koha	Evergreen
	Video Conference	Apache OpenMeetings	Ekiga
	E-learning	Moodle	Sakai
	Directory Services	OpenLDAP	
	Graphics Applications	GIMP	Dia
	Audio/Video Applications	VLC, Movie Player	Rythmbox, Amarok
	PDF Reader	Evince	Okular
	PDF Creator	Libre Office	Open Office
	DVD/CD Burner	Brasero	K3B
	File Compression	7Zip, File Roller	Gzip, Tar
	Document Scanning	Xsane	Simple-Scan
	Vector Image Creation	Inkscape	Libre Office Draw
	PDF desktop publishing	Scribus	OpenOffice.org / LibreOffice
	Postscript view	GNU GV	Evince
	Mail Client	Thunderbird , Icedove	Evolution, Kmail

	Address Book	Evolution	KAddressBook
	Text Editor	gEdit	Kate
	Console Text Editor	Vi , emacs	Vim
	Chatting (Audio/Video)	Empathy, Pidgin	Kopete
	Image Viewer	Eye of Gnome	Gwenview
	File Transfer	Filezila	Gftp
	Printer Management	CUPS	
	3D Creations Tools	Blender	K-3d
	Remote Management	VNC, RDP Vinagre	grdesktop
	Backup Software	Bacula	
	Network Monitoring Tools	Nagios	
	Antivirus	Clamav	
	FTP server	vsftpd	
	Email Server	Postfix	Sendmail
	Proxy server	Squid	
	Web Server Statistics	AWStats	Webalizer
	Blog Engine	Wordpress	
	Wiki	Mediawiki	
	Spatial Database	PostGIS	
	Project Management	DotProject	Redmine
	Issue tracking System	Trac	MantisBT
	Network Security Tool	Nmap	
	Calendar	Lightning	
	CRMApacheOfbiz		
	Diagram Creation	Dia	

3.23 Annexure-III SWOT Analysis of OSS Adoption[29]

Strengths, Weaknesses, Opportunities and Threats (SWOT) Analysis of OSS Adoption are explained in this section.

Strengths

The strengths of adoption of OSS and the potential benefits are highlighted below;

(a) **Freedom to Use & Reuse** Open Source licenses do not limit or restrict who can use the software, the type of user, or the areas of business in which the software can be used. Therefore, OSS provides a licensing model that enables rapid provisioning of both known and unanticipated users.

Because OSS is free from per user or per instance costs, there is a guaranteed freedom to use. Also re-use is enabled.

(b) **Cost Effective** Public agencies get great value and the desired RoI (Return on Investment) from OSS based software-solutions.

(c) **Help Innovation** It is easy to do pilot study and initial roll-outs using OSS with minimal acquisition cycles and associated entry costs. If required, CSS agencies can also be engaged to build value-added capabilities and innovations on top of OSS based software-solutions.

By virtue of their collaborative design, many user-facing OSS based products are intuitive.

Lower barriers to entry, widens participation. OSS is particularly suitable for rapid prototyping and experimentation, where the ability to "test drive" the software with minimal costs and administrative delays is required. CSS suppliers may also provide the same through a 'proof of concept' phase at minimal or no cost; but this approach includes lot of restrictions for use in other phases.

(d) **Better Source Code Level Security** Increased confidence on the software due to the minimised mistrust on the code.

(e) **Better Local Capacity Building**Increased local capacity building for software

[29] Plone-CMS - Customer Segments - SWOT Analysis, 2008 (https://plone.org/events/2008C-summit/customer-segments-swot-analysis#Government)

development based on OSS which leads to effective participation of local industries.

(f) **Preservation of Foreign Exchange** Most of the CSS is imported and hence it drains foreign-exchange. The local support service is, in general, used for OSS adoption which will help local economy to grow; at the same time it helps to conserve foreign exchange and reduction of imports.

(g)**Minimised Piracy** Avoidance of piracy and Intellectual Rights issues which are common with the Proprietary Technology

(h)**Community Support** Availability of Community Support is a key factor for adoption.

(i) **Collaborative & Distributed Approach** This approach is used for developing OSS which has better governance structure.

(j) **Better Interoperability** No vendor monopoly allows use of free and Open Standards. With Data transferability and open data formats, there are greater opportunities to share data across interoperable platforms. Adoption of OSS enhances the interoperability with other e-Governance Solutions because of reuse of recommended software stacks, libraries / components..

(k)**Enhanced Competition** OSS can be operated and maintained by multiple suppliers encouraging competition and providing an opportunity for SMEs to compete in the Government market. This leads to code sharing cultures, better citizen accessibility, and greater control over IT projects. It reduces dependency on a particular software developer or supplier. It also means diversity of support and services choice.

(l) **Growing and mature developer ecosystems** The numbers of community-developers and their quality / expertise are increasing for the popular OSS solutions. Hence, proprietary vendors also initiated their own OSS solutions.

(m) **Rapid and effective vulnerability remediation**[30]The reported vulnerabilities are fixed immediately, in general for the popular OSS solutions.

[30] The Power of Open Source Collaboration Increases VistA EHR Security http://osdelivers.blackducksoftware.com/2013/12/02/the-power-of-open-source-collaboration-increases-vista-ehr-security/

Weaknesses

The weaknesses are existing challenges which are to be considered while adopting OSS; ways to mitigate the weaknesses are also highlighted below;

(a) **Informal use of OSS** There are varieties of OSS solutions available for each domain area. Identifying, selecting and deploying a solution is not a simple task. No recommended OSS Stacks with ecosystem exist but informal use of Open Technology mainly prevails based on the preference / convenience / exposure of the project teams; this adversely affects

i. Maintainability

ii. Security

iii. Bug-fixing

iv. Interoperability & Sustainability

v. Absorption of Technology by Experts

vi. Lack in level of expertise on identified Technologies

vii. Compliance to Security

viii. Sustainability of implementations

ix. Ecosystem

x. Related Intellectual Rights and Legal issues

The OSS Stacks are to be identified and notified for the adoption & reuse with support services in a formal way to mitigate the risks of informal use of OSS.

(b) Adverse Impact of legacy systems Mostly legacy infrastructure and expertise are used. Hence, policy makers and technical experts prefer to continue with the legacy systems supplied by the proprietary vendors. At the same time, there is a little awareness among the decision-makers from public agencies regarding the potential benefits of Open Source and ways to overcome the issues faced during the adoption of OSS. Further, Government organisations are locked with long-term (like 5/10 years) conventional contracts / deals on procuring the ICT systems. This gives little choice for the entry of new systems (which may be based on OSS)..

(c) Limited commercial promotional efforts Since the source code of the Open Source solution is available to all, any marketing done by one company to promote that Open Source solution will also benefit all its competitors. This leads to limited commercial efforts from the industry to promote the benefits of Open Source. Hence there is lesser business motivation from the industry. The Government needs to initiate the promotional efforts and awareness on the adoption of OSS. Industry could consider promoting the OSS based on the better quality of their services.

(d) **Dominance of existing Marketing Forces** In general, major ICT suppliers are preferred over SME (Small and Medium Enterprises) in Government procurements; hence majority (about 80%) of the ICT procurements are done with a few (about 10 or less) business establishments. Most of the major ICT suppliers generally prefer the use of CSS (Closed Source Software) because of their long-term business tie-ups with OEM(Original Equipment Manufacturers) of CSS. Majority of OSS solutions are provided by Small and Medium-sized Enterprises (SMEs) in most of the countries. The vast majority of Government IT work is still carried out by the major ICT suppliers resulting in lesser participation of SMEs.

(e) The existing marketing forces create fear, uncertainty and doubt about new entrants from Open Source model in order to avoid further competition. Hence, entry of new participants from Open Source model needs support from Government to have a level playing field between Open Source and CSS.

(f) **High Cost of Integration and Migration** Most of the existing proprietary systems poorly inter-operate with other software; this is done mainly to retain the customers.; Cost of switching from existing CSS to other OSS becomes extremely expensive. OSS would require additional developmental efforts to enable integration with an existing proprietary environment. Some OSS never works well with established proprietary products. Hence Government guidelines are required to avoid lock-ins; solutions which offer standards-based interfaces should be preferred.

(g) **Security Issues** The availability of source code makes the OSS vulnerable to more threats. However, this should be mitigated by using the recommended stable version of OSS with necessary support & updates.

(h) **Lack of OSS Policy / Framework** In-spite of many potential benefits & promises from the Open Source, the Government intervention, through Policy / Framework on OSS (like UK, European Union), is still needed. The proposed Framework would mitigate the issues due to lack of OSS Policy / Framework.

(i) **Lack of cost effective Support Services** Sometimes, support and maintenance costs outweighs those of the proprietary package and include 'hidden' commitments. Sometimes adequate support may not be available and it becomes biggest weakness of OSS. Support on voluntary community basis alone may be insufficient. A full assessment of the total cost of ownership along with the support service costs from the supplier will help to mitigate this risk.

(j) **Lack of Motivation, Capacity Building and Awareness** Government staff are traditionally trained (and practised) in using CSS programs, the introduction of new programs / software may require staff retraining in order to enable them to use OSS. It is often assumed that OSS requires specialised skills – not necessarily programmers – but usually a systems administrator type of person to configure the application. Institutions change slowly – change takes time and it often makes people nervous.

(k) **Lack of awareness on TCO**The lack of awareness on the total costs associated with the adoption of OSS is another common problem. The provisioning of simplified & customised TCO model would mitigate risks.

Opportunities

The opportunities provided by the adopting OSS and the potential benefits resulting from the opportunities are highlighted below;

(a) **Low Barrier to entry** OSS introduces very low barrier to entry compared to the CSS whose prices are mostly increasing every year. OSS coupled appropriate hardware (whose prices are falling every year) offer a lot of scope for the wide spread adoption in e-Governance systems.

(b) **Economic Opportunities for Local Industry** Hardly few Indian CSS are available and hence their impact on the Indian economic growth is negligible. Whereas, OSS offer many new business opportunities to local industry in the form of offer of support services on OSS, capacity building on OSS, innovation of new products (including OS, Cloud, VM, solutions, Applications) using OSS libraries, development of integrated solutions on desktop, server, embedded, cell phones, set-top boxes, network, open hardware (like 3D printer, robot), etc.; thus OSS provides more growth opportunities to local industries (including SME, start-up companies).

(c) **Wider choices on OSS** There are many competing support-service providers on the OSS solution, in general; this is in contrast to limited choices with the case of CSS where one company along with their partners are offering support services; hence multiple options are available with OSS solution to choose suitable service-agency; this leads to simpler & cost-effective approach in case of switching the support service agency.

(d) Similarly, many OSS competing distributions are also available for core areas like OS (Ubuntu, Debian, BOSS), database (PostgreSQL, MySQL), web server[31] (Apache, Nginx), application server(Tomcat, JBoss, Jetty), etc.; in case of migration requirement, moving from one OSS solution to another OSS solution is comparatively easier due to their openness.

(e) **Ability to drive cross-industry collaboration** it facilitates the cross-industry collaboration through consortia like Linux Foundation, OpenMAMA, etc.

(f) **Forking** Sometimes forking of OSS solution occurs for good reasons; for example, MariaDB is a community-developed fork from the company controlled MySQL database. Similarly, Proprietary Unix implementations (like SCO, Solaris, IRIX, HPUX) were forked into OSS BSD versions (Open BSD, NetBSD).

Threats

The threats are potential challenges to be considered while adopting OSS and ways to overcome them are highlighted below;

(a) Decision-Makers Slow change of perceptions of decision makers of e-Governance Systems about OSS solutions. License Model, Intellectual Rights Infringements and Legal compliance are often misunderstood. Conducting awareness programs and provisioning of appropriate reports will help to take better decisions.

(b) Resistance The status quo of the established institutions is threatened by the new entry of OSS; hence, fears, uncertainties and doubts (FUD) are created by the established institute to retain their hold on users by creating incompatible solutions (like interfaces, device-drivers, patents) with the established proprietary solutions. This can be minimised by the Policy / Framework on OSS and its enforcement in e-

[31] September 2012 Web Server Survey: http://news.netcraft.com/archives/2012/09/10/september-2012-web-server-survey.html

Governance Systems.

(c) **Support Services** Non-availability of support services with adequate guarantee is a potential threat while adopting OSS. Use of wide variety of OSS solutions for the specific domain area makes it difficult to engage support services. The approved OSS Stacks and provisioning of support services will improve the situation.

(d) **Activity** Lack of continued development of OSS solution is another threat to be considered. Sometimes, the dependency library may be missing or available only on proprietary model. The approved OSS Stacks will improve the situation.

(e) **Incompatible Versions** Sometimes there may be mismatch among various libraries of the integrated OSS solution. The approved OSS Stacks will ensure the compatibility.

(f) **Staff** Sometimes there may be a lack of sufficient number of in-house experts on OSS and need for more skilled staff when OSS is used. There are lesser incentives for the migration to OSS systems. Capacity Building and Policy / Framework on Adoption of OSS will improve the situation.

(g) **Risk of forking**The forking occurs mainly due to developers who try to create alternative means for their code to play a more significant role than achieved in the base OSS solution. The approved OSS Stacks will reduce the risk.

(h) **Absence of OSS implementation Agency** This scenario hinders economic and technology opportunities for the industry.

3.24 Annexure-IV Common Influencing Factors for the Adoption of OSS

Technological Level Factors

(a)**When to comply of Device Drivers for OSS OS (+)** - The availability of device drivers for the GNU / the Computers and associated peripherals procured would be operational on GNU / Linux OS. Thus availability of device drivers enhances the adoption of OSS.

(b)Technological Compatibility based on Standards (+)- Better compatibility / interoperability enhances the chances of adoption of any software. Insisting on adherence to Open Standards & Data Formats (instead of insisting on compatibility with legacy systems) is the better-way for the compatibility.

(c)Technological Complexity in OSS usage (-)- Complexity reflects the ease & simplicity of OSS in understanding and usage. More the complexity, lesser the adoption. The provisioning pre-configured & bundled OSS Stacks with adequate support would mitigate the issues due to complexity, if any.

(d)Relative Advantage of OSS (+)- OSS has an added advantage due to reliability, scalability, ease of use, functionality and security from virus attacks and spam etc.; this leads to reduced TCO.

(e)Trialability of OSS (+)- The degree to which it is possible to use OSS for proof of concepts and experimental studies.

(f)Presence of Proprietary Lock-in (-) - The more lock-in with the legacy/new CSS creates more barriers for the adoption. The proposed Framework on OSS would minimise the proprietary lock-ins.

(g)**Freedom to modify and improve (+)** - This makes OSS more suitable for customisation and enhancement as per requirements.

Organisational Level Factors

(a)**Management's Positive Attitudes towards OSS (+)** - The attitudes & support of the Senior Management towards provisioning rules, training, support services, provisioning of additional resources (hired manpower / consultants, hardware and network facilities) for the OSS adoption, considerations on strategic importance are very crucial. Better attitude affects the adoption positively.

(b)Champions of OSS (+) - A combination of clear long term plan for training & support services on OSS and availability of champions of OSS in the senior management, in an organisation creates very powerful impact on the adoption.

(c)Size of Organisation (+) - The size of the Government organisation indicates the numbers of Government employees working. Large size generally facilitates better adoption. But even smaller size also facilitates if the better awareness is available about the benefits of OSS with the stake-holders. The awareness programs would help the adoption.

(d)Diverse Expertise at Management Level (+) - The wide variety of competence of Senior Management towards OSS. More competence means better chances for adoption, in general.

(e)Level of Formalisation (-) - The level of formalism and bureaucracy in the organisation. High level of formalism mostly inhibits the adoption. However, if OSS is accepted as part of formal procedures, then the formalisation facilitates its adoption.

(f)Centralisation on Decision Making (-) - The decision-making power being concentrated with only few experts in the Senior Management affects the adoption negatively, in general, in the initiation phase and positively in the deployment phase. However if these experts are aware of the benefits of OSS, then centralisation also facilitates the adoption by overcoming cultural and structural barriers.

(g)Inter-connectedness of Organisation (+) - The level and depth of connections among various units of the organisation. Better connectedness mostly facilitates the adoption.

(h)Organisational Slack on Resources (+) - The availability of internal resources of the organisation that are not yet assigned with specific works but can devote their time for new works on OSS. The larger the availability, better the chances for the adoption.

(i)Inclination towards Business Processes Re-engineering (+) - More chances for change-procedures / business-processes re-engineering in the organisation offer better adoption.

(j)Availability of Internal Technical Expertise (+) - The technical expertise on OSS available in the organisation impacts the adoption positively. The involvement of in-house experts through collaborations and capacity building through awareness program & training would enhance the expertise of in-house experts.

(k)Availability of Financial Resources (-) - The limited financial resource (shortage of budgets) availability in the Government organisation enhances the adoption. New metrics are required to give more weight-age for the project plan which results in better saving and wider reuse of ICT assets.

(l)**Outsourcing impact (-)** - The reduction/elimination of in-house experts due to outsourcing would reduce the adoption, in general.

Environmental Level Factors

(a)**Rules for the adoption of OSS (+)** - The rules facilitate the adoption of Government's OSS policies and guidelines. More rules mean better chances for the adoption. The rules should be applicable to all levels of employees. However, rules with long term contract with CSS would hinder the adoption.

(b)Provision for Capacity Building (+) - The level and availability of awareness programs & trainings on OSS for the adopters of OSS are very crucial factors. Better level reduces the barriers for the adoption.

(c)**Availability of Support Services on OSS (+)** - In case of a bottleneck or failure of a system based on CSS, then it is possible to hold the vendors of CSS; whereas, the project team or champion / mentor has to own the responsibility when the project is based on OSS.

Hence the availability of external support, especially for services such as the installation, configuration and maintenance of OSS, is a very crucial factor. The adopters of OSS are more willing to pay for support.

The proposed Framework on OSS, pre-configured & bundled OSS Stacks and better assured long-term support services with SLA enhances the adoption and minimises the liability on the project team / champion / mentor.

(d)Competitive Pressure (+) - Early adoption of OSS by the competitors enhances the adoption.

(e)System Openness (+) - Indicates how much the organisation is possibly considering suggestions towards OSS from external environments? Higher the openness, better the chances for the adoption. At the same time, poor adoption of OSS in other external organisations hinders the adoption.

(f)Past Experience on OSS (+) - Success case studies on OSS adoption, past experience of the OSS users / developers and showcasing them create more confidence on OSS.

(g)**Availability of Internal Collaboration Mechanism (+)** - The availability of collaborative information mechanism within the Government like discussion forum enhances the adoption.

Individual Level Factors

(a)**Level of Organisational Objectives Consensus (+)** - The level of clear understanding among the adopters of OSS about the organisational objectives, their agreement & motivation. This may require more efforts for the adopters to learn about OSS. Lack of motivation hinders the adoption. Better consensus enhances the adoption. This may require more awareness programs on OSS.

(b)**User's Fear on De-skilling of Legacy Expertise (-)** - The fear of users to become deskilled by losing their expertise in popular legacy proprietary systems while migrating to OSS.

Some users have perception that their work would be under-valued if they use OSS; since most of the project evaluation rating consider more value if more project expenditure; the saving in project expenditures and its impact in reusing the system (based on OSS) without additional cost are not considered in general. Some fear that high level of technical expertise is required for the use of OSS. All these fears create barriers for the adoption. Government rules and promotional drives for OSS reduce the fear and create confidence on OSS.

3.25 Annexure-V Guidelines for Establishing Enterprise security with OSS

(a) Protect network with a strong firewall - A security hardened Linux distribution (OSS like Smooth wall) which provides critical hardware firewall operations like port blocking, IP blacklisting, antivirus protection, etc. can be considered; at the same time, it should be easy to use.

(b) Secured Remote Access - Many times, it is required to work through a secured solution (OSS like Open VPN) from remote places with an access to office/data-centre resources. The solution should work on major platforms with localised control and GUI for easy use.

(c) Securing Data on local desktops & laptops using encryption - There is a risk of exploiting the sensitive data residing in local desktops and laptops by unauthorised persons. The common recommended solution is to use encryption solution (OSS like True Crypt) so that even if there is a physical access of the local system by unauthorised persons, the content cannot be used without the required digital key.

(d) Securing Wi-Fi access points - The Wi-Fi access points are required to be protected by using appropriate solution (OSS like WPA2 with RADIUS authentication server) to have safe network for the organisation; the solution allows the authorised users to login easily with username and password while hiding its encryption keys from the end-users.

(e) Adopting Best Practice for System Administration - All users should use strong passwords. Multi-factor strong authentication should be enabled with the combinations of One-Time-Password (OTP), Digital Signature, Finger-Print biometrics, etc. If same authentications are to be repeated in multiple applications, then Single-Sign-On (SSO) authentication solution (like Central Authentication System - CAS) can be used. Only the required services should be invoked in the systems especially at the data-centre; that is, the solution which is not required for running the current system should be turned off. Similarly, monitoring the logs and file folders should be done using appropriate solution (OSS like Mon) for any suspicious activity on regular basis; automated alerts and polls can be activated. Appropriate backup and disaster recovery mechanism (local / remote locations) are to be enabled. Similarly, creations of logfiles at the application level are to be enabled at remote servers.

(f) Secured Internet Access from Intranet through Web Proxy - A web proxy (OSS like Squid) should be made available to route, filter-out & monitor the web access and also to prevent the downloading of mal-ware.

3.26 Annexure-VI Adoption of OSS – International Scenario

The initiatives taken by various public agencies / Government world-wide are outlined in this section.

Promotion through Policies - OSS promotion strategies via Government procurement fall into four broad categories[32]

Mandating OSS

Preferring OSS

Mandating Open Standards

Best value

Major International Promotions

European Union Initiatives - European Commission (EC) published a report about avoiding vendor lock-ins in Government ICT systems[33] along with an ICT Procurement Guide based on ICT Standards and Good Practice. It is expected to enable more interoperability, innovation and competition, lowered costs (by more than 1 billion Euros per year), and improve interaction with citizens.

European Commission (Join-up program[34]) has decided to join hands with Australia (Open Ray program), New Zealand (Open Ray program) and Vietnam (Open Road program) to enhance the software solutions by sharing and reusing. Join-up hosts more than 300 OSS projects directly now and hosts more than 4,000 projects in collaboration with other communities / forges in European Union.

Laws on the adoption of OSS in e-Governance were brought out by European countries like Italy and Iceland.

[32] UNDP-APDIP - Free/Open Source Software - Government Policy, http://www.iosn.net/government/foss-government-primer/foss-govt-policy.pdf

[33] Against Lock-in in ICT Systems, 2013, http://opensource.com/Government/13/7/against-lock-in-ICT-systems

[34] Sharing and Reusing of OSS, https://joinup.ec.europa.eu/community/osor/news/australia-new-zealand-vietnam-and-ec-coalesce-platforms-sharing-and-re-use

USA - Department of Defence[35] (DoD) has large number of applications based on OSS and has been implementing a roadmap to adopt OSS and Open Standards, as such a move is not only in the US national interest, but also in the interests of US national security. The time-line of the major-events, publications, and code releases in the history of the US Government's adoption of OSS is also available[36].

France [37] **-** French Government issued a guideline [38] , to "systematically review" alternatives to CSS when obtaining or developing new versions of applications; it also recommends to build internal expertise on OSS, pooling of resources, collaborating with OSS communities, and contribute back to OSS projects. OSS solutions are widespread in Government organisations; about 15% of country's IT budget is spent on services related to OSS and this trend is increasing. A new law[39] on giving priority to OSS in Higher Education and Research was brought out by French Parliament.

The reasons for the major success of France in the adoption of OSS include[40]:

Smaller OSS companies have effectively organized themselves into alliances and are growing into pure Open Source consortia, which have helped them access the legal expertise to participate in tenders and to better educate policy makers and ICT (information and communications technology) professionals.

France has the largest Open Source market in Europe and demand for Open Source from public agencies is high.

The French government actively supports Open Source R&D projects through so-called "competitiveness clusters," which consist of large, medium, and small companies, as well as academics.

The government at the highest level not only encourages administrations to consider Open

[35] Open Technology Development - Lessons Learned & Best Practices for Military Software http://www.oss-institute.org/OTD2011/OTD-lessons-learned-military-FinalV1.pdf

[36] Open Source in the US Government http://gov-oss.org/.

[37] Sharing and Reusing of OSS, https://joinup.ec.europa.eu/community/osor/news/australia-new-zealand-vietnam-and-ec-coalesce-platforms-sharing-and-re-use

[38] OSS-Guidelines, https://joinup.ec.europa.eu/news/french-guideline-favours-use-free-and-open-source

[39] Free Software Law for Higher Education in France, July, 2013 https://joinup.ec.europa.eu/community/osor/news/french-parliament-makes-free-software-law-higher-education

[40] Case study of Open Source Policies and Implementation, 2013 Jan, https://joinup.ec.europa.eu/news/inertia-hindering-governments-profit-open-source-benefits

Source, but now also allows savings realized through Open Source deployment to be used to invest in in-house OSS expertise and participation in upstream projects.

A conducive infrastructure, adequate tender laws and policies / guidelines, policy makers' support & provisioning of additional resources, awareness among the implementers are available for successful implementation of OSS.

UK - The Government of the United Kingdom'[41]wants to create a competitive software market, where OSS and CSS compete on an equal basis; it wants to avoid lock-ins by making long-term commitments to any particular technology, product or supplier; this ensures maximising the future development options and avoid technology lock-in if at all possible. Open Source Procurement Toolkit[42] is also made available by UK Government.

UNDP Initiatives - UNDP has taken many initiatives for promotion of OSS and bringing many important reports / guidelines on OSS. The International Open Source Network[43] (IOSN) is an initiative of UNDP's Asia Pacific Development Information Programme (APDIP) and operates under the principle of “Software Freedom for All” (SFA). Its work includes provision of support and assistance, centre of excellence and information house for OSS in the Asia Pacific region. Through the IOSN/SFA initiative, UNDP provides policy support and advisory services to Government bodies, non-profit organisations and others.

Recognising India’s strength in OSS, UNDP/IOSN has selected C-DAC of DeitY, Government of India, as its South Asia node.

China - China brought out office document format known as Uniform Office Format or Unified Office Format (UOF) in 2005 and later RedOffice was implementation was also developed based on UOF.

In the 11th Five Year Plan (2006–2010), OSS policy was announced. The use of foreign software in Government Offices was discouraged; the locally packaged OSS systems are preferred as local software. China brought out its own Linux distribution known as "Red

[41] UK Government Service Design Manual, 2013, https://www.gov.uk/service-manual/making-software/choosing-technology

[42] UK OSS Procurement Toolkit https://www.gov.uk/government/publications/open-source-procurement-toolkit

[43] IOSN, http://www.iosn.net/

Flag" as an alternative to Windows. As per a paper "The Emergence of Open-Source Software in China[44]", 2007, Red Flag held 30 % of the desktop market in China.

The adoption rate of smart-phone with Android Linux OS is about 90% in 2012. Almost all Super Computer and Cloud Data Centre are based on Linux OS. In 2013, China announced that it is bringing out another Linux OS based on Ubuntu in collaboration with M/s. Canonical, UK.

OS China[45] is similar to Sourceforge source code hosting service; it hosts about 24,000 projects and many Chinese developers are contributing back. The latest release of the Linux kernel includes about 11,000 contributions from Chinese developers, according to Black Duck's research (2013).

[44] The Emergence of Open-Source Software in China, http://www.irrodl.org/index.php/irrodl/article/view/331/762

[45] OS China, http://oschina.net/

3.27 Annexure-VII Adoption of OSS – Indian e-Governance Scenario

At present the FOSS movement in India has begun to gain mainstream acceptance and the initiatives taken by Government of India given in this section.

FOSS Cell, DeitY initiatives on FOSS

DeitY established FOSS Cell in year 2004 for promotion of FOSS in the country and has taken number of key initiatives creating an eco-system; the major one is setting up of National Resource Centre for Free & Open Source S/W (NRCFOSS) through C-DAC, Chennai.

Adoption of OSS in e-Governance Projects at Different States

A number of State Governments have started to adopt Linux and Open Source Software as their defacto platforms for e-Governance applications deployment.

Kerala: State Government of Kerala has decided to use OSS for the e-Governance and IT education in the schools. Kerala's draft IT policy focuses on e-Governance, Open Source software and development of technologies. Major proposals in the state include establishment of an International Centre for Free Software and Computing for Development, ITES Training Centre (in Kochi) and extension of Internet to all educational institutions and villages by 2010. Open Standards such as Unicode and Open Document Format and Open Architectures will be followed in e-Governance projects to avoid vendor lock-in. ICFOSS(International Centre for Free and Open Source Software) is an autonomous institution under the Government of Kerala with the objectives of coordinating FOSS initiatives within Kerala.

Tamil Nadu: Tamil Nadu is actively pursuing the implementation of OSS. Electronics Corporation of Tamil Nadu (ELCOT), adopted OSS in May 2006 and the entire ecosystem at ELCOT is build around OSS. Tamil Nadu Government can save Rs 200-500 Crores every year through National e-Governance action plan. Some of the OSS solutions that have been developed for the Government include: Anywhere property registration software, Old age pension software with a public interface, Office file management software, and Web-based land recovery administration software. ELCOT has also developed software for the disabled called ORCA based on Ubuntu. ORCA is a text to voice software developed for people who are visually impaired.

Uttaranchal: In a significant move towards promoting e-Governance in India, the

Government of the Indian State of Uttaranchal has signed two Memoranda of Understanding (MoU) with IBM, to mark the beginning of a State-wide e-Governance and University Programme initiative. This is the first implementation of IBM’s e-Governance framework in India. Based on open-source technologies and Open Standards, IBM’s e-Governance framework enables interoperability between new and existing applications.

Assam: The Assam Government has issued an OSS policy to promote use of FOSS in all the Departments and State agencies, bodies and authorities and imparting training on FOSS in schools and colleges. The Government Departments and bodies would ensure that Open Document Format (ODF) is adhered to in creating and storing editable documents, data and information and all applications developed by the respective Departments adhere to ODF and other Open Standards and are largely independent of Operating Systems and web browsers and any generic hardware procured has support for multiple Operating Systems such as Unix, Linux, Opensolaris and other Open Source platforms.

West Bengal: TheITDepartment of West Bengal government is choosing Open Source operating systems for its ambitious e-Governance programme in the state. Government has chosen to use Linux for various e-Governance programme involving 277 panchayats in Burdwan district. The IT Department has set up a computing centre which operates exclusively on OSS.

Besides above, other states in India are also showing keen interest in OSS solutions. Union territory of **Pondicherry** was among the first regions to adopt OSS. Many of the Department portals like Commercial Taxes Dept, Transport Department have been developed using OSS.

Haryana Government has signed an agreement with Sun Microsystems to use Sun's Open Standards-based productivity package, StarOffice 7 Office Suite, across all State Government Departments.

Adoption of OSS in e-Governance Projects by NIC, DeitY

Some of the e-Governance projects based on OSS are listed below; most of the projects mentioned below are using **PostgreSQL** as the database.

JAVA Technology: eOffice Project , e-Procurement system, Vimanic Pilot Examinations Application for DGCA , Sarathi – Driving License, Vahan – Registration of Vehicles, Common Integrated Police Application (CIPA) , CIPRUS Project, Immigration Visa Foreign Registration Tracking (IVFRT), ePDS, National Minorities Scholarships Project, Multipurpose National Identity Card Software Project, Karnataka Judiciary Department Application, Karnataka Administrative Tribunal, Karnataka Employment Exchange Project, Sevarath Payroll application, TreasuryNet application, CollabCAD, Collabland, TWADNEST, e-District and PDS allotment distribution & monitoring Systems.

PHP Technology: District Court Information System Software, e-Courts , Defence Land Records Software Project (RakshaBhoomi), DC-Suite, Below Poverty Line (BPL) Software Project, NREGA, Online Local Bodies Election of TN, Portal for Rural Development Dept, Specimen Status Monitoring Systems for Forensic Sciences, Utility Maps Web-Interface

Application Portal based on Drupal : Central Public Procurement Portal, NIC-OTC, NIC-Pune about 10 Portals, NIC-SDP, Transport Dept. of Arunachal Pradesh, About 50 Portals of various Departments of Karnataka state by NIC-KASC and State Portal based on Drupal – Tamil Nadu, Meghalaya, Tripura

Plone Technology: IntraNIC, IntraGov, IntraYojana, IntraMHA, IntraDIT, IntraHealth, IntraPIB, IntraCA, IntraPMO, IntraPOWER, IntraORISSA

OSS Servers at Data-Centres of NIC, DeitY

The following table shows the usage of OSS at Server Level (Including Virtual Machines) in various e-Governance projects developed, hosted and maintained by NIC at the National Data centres and NIC State Data Centres as on July, 2013.

S.No	Description	Percentage Deployments
1	Linux Physical Servers (including RedHat, CentOS, Debian, Ubuntu, BOSS, SUSE etc.)	32 %
2	Windows Physical Servers	65 %
3	Other OS Physical Servers (including Solaris, IRIX, etc.)	3 %
4	Linux Virtual Machines	69 %
5	Windows Virtual Machines	31 %

Open Technology Centre Project

OTC (https://portal.otc.nic.in/) is a Project, sponsored by DeitY, MCIT, Government of India, implemented by Open Technology Group (OTG), NIC. OTC Project is spearheading identification as well as adoption of Open Technology in e-Governance applications and services managed by NIC/NeGP for both State and Central Government Agencies.

Key Technology domains supported by OTC are Drupal CMS/Portal, SymmetricDS Database Replication, Database Migration to PostgreSQL, CAS Single Sign on Solution, Verification Services based on 2D Barcode, Platform independent Digital Signature Certificate, Recommendation and support provisioning of OSS Stack, Bundled OSS Stack for Development, Staging & Deployment ,offering of VM Service, Capacity Building & Hand holding, eForms using HTML5 / Xforms and Performance Tuning of Open Source Application Servers.

OTC has set up collaborative infrastructure (using Portal, Issue Tracking System) for supporting its activities. OTC has evolved a multiple-level support model for the identified OSS Stack.

FOSS initiatives at C-DAC, DeitY

DeitY has taken FOSS initiatives, like NRCFOSS, BOSS-GNU/Linux, Meghdoot-Cloud through CDAC to adopt and promote OSS.

NRCFOSS (www.nrcfoss.org.in.) was setup in Chennai during April 2005 with the twin roles of bridging the digital divide as well as strengthening the Indian Software industry. NRCFOSS contributes to the growth of FOSS in India through Research & Development, Human Resource Development, Networking & Entrepreneurship development, as well as, serve as the reference point for all FOSS related activities in the country.

Phase – I :NRCFOSS introduced proof of concept based FOSS Technologies in the formal & non formal sectors like engineering undergraduate curriculum of the Anna University with an affiliation of 254 engineering colleges to train teachers of engineering colleges and equip them to offer FOSS electives and student projects in their academics (UG/MCA levels) as part of the curriculum aiming for successive graduated engineers with exposure, training and skills in FOSS technologies.

Phase-II : This isa consortium of C-DAC, Anna University (AU-KBC Research Centre) and IIT-Madras, IIT-Bombay mooted and approved by DeitY with the following objectives:

(a) Development of SaaS stack delivery model in area like Grid Computing / Cloud Computing

(b) Integration and development of Common desktop development infrastructure

(c) To setup Centre of Excellence for Mobile Internet Devices based on BOSS Linux

(d) Creation of NRCFOSS centralised portal for involvement, analysis, R&D and knowledge exchange

(e) FOSS HRD in the formal & Non-formal sectors

(f) Creation and maintenance of knowledge bank repository for education, e-Governance & scientific applications.

In continuation with the work done by AU-KBC Research Centre through the phase-I of the project, I.T curricula has been enhanced FOSS theory and practical sections. Some of the Universities / Colleges who adopted FOSS as elective in their curriculum are Anna University, Loyola College, Chennai, Rajasthan University of Technology. Anna University is offering online course MSc (FOSS) The details can be seen at http://cde.annauniv.edu/MSCFOSS.

BOSS GNU/Linux (Bharat Operating System Solutions – http:/www.bosslinux.in) is a Desktop and Server Linux Operating System with Indian language support derived from Debian Linux developed by C-DAC, Chennai. Also BOSS is customized (EduBOSS) for the ease of use in schools and colleges across the country.

BOSS Support Centre Network: BOSS Linux support Centres project have been setup at various C-DAC Centres. Franchisees have also been used as part of the support centre network. In addition, a National Help Desk facility setup at C-DAC Chennai also provides the additional layer of support. Many State Governments and National institutions have adopted BOSS Linux; some of them are Punjab, Haryana, Tamil Nadu, Chhattisgarh, Tripura, Kerala, and Pondicherry. Indian Navy, Indian Army. Promotional and awareness workshops are conducted across the country. Over 250+ colleges across the country have labs with BOSS Linux installed. Efforts are being taken to bring vendors on board to create an eco-system for BOSS Linux.

Business Model: The Business Model adopted for the BOSS Linux promotion is the Services and Support strategy. License for BOSS is free and the service and support are charged. The revenue comes from branding, training, consulting, custom development, and post-sales support instead of traditional software licensing fees. This could be in a subscription mode charged nominally per desktop per year or is charged lump sum towards provisioning of on-site support. C-DAC has tied up with various vendors to provide technology support on preloaded BOSS Linux on desktop/laptops with minimum price.

In addition to above direct revenue earning, BOSS Linux adoption by the various Government agencies / Departments has resulted in an indirect savings to the Government.

MeghdootC-DAC has also developed a cloud product called Meghdoot which offers various features in cloud environment such as Platform and Infrastructure as a service (PaaS and IaaS), On demand dynamic provisioning, Metering & Monitoring, Graphical Installation of Middleware stack, Web based Management of Cloud resources, Provision for deployment of multi instance user appliances, Customized Elasticity, Web service based management of cloud, High Availability, Enhanced Security across layers. Meghdoot Cloud Stack has been deployed at the Tamil Nadu State Data Centre, CHiPS Chhattisgarh, and Indian Navy.

3.28 Annexure-VIII Rating of OSS against other CSS using Business Metrics

Basics

The business metrics are needed to identify & demonstrate whether OSS is cost-wise also superior compared to CSS. These metrics along with other technical points influences the decision-makers to take appropriate decision whether to opt for OSS or CSS while developing e-Governance systems.

If the OSS solution is to be evaluated against CSS, then models like

(a) Return on Investment (ROI)

(b) Internal Rate of Return (IRR)

(c) Total Cost of Ownership (TCO)

can be considered. If required, these models need to be analysed to select / customise a suitable model.

Approach for Return on Investment

ROI find outs the financial performance of an investment by evaluating the efficiency of the investment; it includes not only the resulting benefits to the organisation due to the investment but also the cost elements.

Approach for Internal Rate of Return

IRR, sometimes is called as Rate of Return (ROR) or Discounted Cash Flow Rate of Return (DCFROR). It indicates profitability of an investment. Higher the IRR, then more value to the investment. IRR is somewhat difficult to understand when compared to metrics like TCO, ROI

Approach for TCO

There are various models used in evaluating the Total Cost of Ownership (TCO). [46]

Simpler Approach The conventional analysis used in Total Cost of Ownership (TCO), in general, simply assumes the total cost involved in the initial procuring (CAPEX) and operating / implementing (OPEX) the particular software. The early TCO studies, in general, did not considered costs like exit/migration costs.

Missing Cost Factors Switching Costs due to lock-ins, may include damages due to contractual commitments, the cost of replacement equipment, loyalty programs, search costs, transaction costs and uncertainty about alternative suppliers, conversion of data & its risks, retraining and compatibility.

[46] "Total Cost of Ownership of Open Source Software" (PDF), London School of Economics (LSE), http://ctpr.org/?p=701

Variations of Cost due to environment In developed countries, where labour costs are high, the relative low support cost of OSS need not necessarily reduce total costs of using and maintaining systems; when labour costs are high, labour-intensive components of the total cost represent a high share of the total cost, making the licence fee itself (which is not present in the case of OSS) less crucial.

In contrast, when labour costs are low in a developing country, the share of the licence fee of the software in the total cost of ownership is much more significant, even prohibitively so; even after software price discounts, the price tag for CSS, in general, is enormous in purchasing power terms. The labour-intensive components of the total cost for the Open Source are comparatively very low in developing country; these expenditures, in general, result in local currency to be paid to domestic industry.

Non-Quantifiable Factors However, there are many factors which are non-quantifiable in terms of cost; for example, enhanced security & reduced mistrust, reduced service disruption, reusing the software, etc.

Alternative detailed Cost Model Some attempts are made recently to account additional costing for some of the above factors. In a report[47], the alternative cost model ("Total Lifetime Cost of Ownership"), including search, exit and transition costs, is recommended. The report says "TCO reflects a measure of all the costs of identifying and acquiring software, away from the software. TCO reflects not just the direct qualities of a software product (price, functionality, reliability) but also the relationship of the software to the organisation's broader set of technology platforms, installed systems, skills and strategic goals, available market and community based services."

Local Economy One also has to see whether the money is given to local ICT industry and if the spent-money helps to preserve foreign-exchange and to grow the local knowledge-base (SME / local Community) with in the country.

Reuse Cost Not only the immediate cost benefits but also the long term benefits, like reuse of ICT assets in other public agencies, self-reliance in ICT knowledge-base, the improved negotiating power of entire Government as a single entity, are also required to be considered.

Conclusion on TCO All these facts suggest that focusing on conventional TCO model alone is not enough. Alternative TCO models, after customisation to suit developing countries, may be considered to see appropriate impact. However, TCO mainly focuses on cost factors and generally misses benefits/returns.

[47] Total Cost of Ownership of Open Source software: a report for the UK Cabinet Office submitted by Shaikh, Maha and Cornford, Tony, London School of Economics and Political Science, 2011, http://eprints.lse.ac.uk/39826/

Typical Total Cost of Ownership (TCO) Estimation				
Cost Category		Cost		
Search	Cost of up-front evaluation study			
	Cost of up-front proof of concept implementation			
	Total Search Cost			
Acquisition	Cost of Software			
	Cost of Customisation for business needs			
	Cost of Integration (to current platform)			
	Total Acquisition Cost			
Integration	Cost of Migration (data and users)			
	Cost of Training			
	Cost of Process and Best Practice change			
	Total Integration Cost			
Use	Cost of Support services - in house			
	Cost of Support services – contracted			
	Cost of Maintenance and Upgrades			
	Software scaling (for change in user or transaction volumes)			
	Total Use Cost			
Retire	Exit costs (in relation to hardware and software)			
	Exit costs (in relation to changeover, re-training)			
	Total Retire Cost			
	Total Cost			

3.29 Annexure-IX Rating of OSS based on Performance matrix

Basics

The basic step for evaluating OSS or CSS are essentially the same. Typically it can follow the four simple steps

i) identify,

ii) review,

iii) compare

iv) analyze.

The amount of effort spend evaluating OSS software is strongly dependent on how complex and important the OSS software is for the organisation.

The quality of OSS solution is affected by many associated variables related to the OSS solution and its stakeholders; the number of variables may be limitless and each variable can be interpreted by others in different ways. Further, the adoption of the OSS solution is affected often by the reputation of the Partner-company / Trust rather than the real quality of the OSS solution. Hence, it is necessary to identify a suitable methodology with a set of structured criteria to access the quality of the OSS solution.

Some of the variables associated with the rating of OSS solution:

a) **Adoptability** - the number of downloads, the number of users / well-known users, awards, books, ease of use, modularity, by-products, etc.

(b) **Activity** – showing the progress made by the developers, road map, the number of bugs reported, bugs fixed, new features and discussions, etc.

(c) **Longevity** – how long the OSS solution has been in use

(d) **License** – is one of the general Open Source licenses used which indicates a set of well-defined conditions for the contribution of code to the ongoing development of the software; the flexibility without restrictions to implement alternative formats, integration between the proprietary solution and other systems, etc.

(e) **Fork-ability** – fork probability based on open model, protection against proprietary forks.

(f) **Services** – quality of support, capacity building and consulting from the community, industry and other paid-models.

(g) **Documentation** – user manuals and tutorials, developer documentation

(h) **Security** – reporting / responding to vulnerabilities

(i) **Functionality** – testing against functional requirements which can be further classified as essential and desirable.

(j) **Integration** – standards, modularity and collaboration with other products

(k) **Nature of the Trust** – the reputation of the Trust which is acting as a driving force behind the project on OSS solution with a very clear development process, level of democracy of management, impact on types of distributions (OpenCore with limited features on open model, Enterprise with enhanced features on proprietary model) released on the OSS by the Trust, etc.

(l) **Skill Set** – the skill sets available in the user/Partner-Company/Developer/Trust of the OSS solution which indicate the readiness of user to adopt the OSS solution, etc.

3.30 Annexure-X Key Stakeholders of Ecosystem

Stakeholder	Roles & responsibilities
Senior Management	Policy / Decision Makers from Government who take decisions on the Projects; they guide Project Management.
Project Managers	Government / Department users who are responsible for the Projects and adopts the policies & guidelines taken by the Senior Management; they supervise the services from registered partners like Product-Partner, Technology Experts, System-Developer, System-Integrator, Service-Provider.
System-Developer	Person / Agency who is assigned with development, deployment and maintenance of systems under the supervision of Project Management; they avail the services from Product-Partner, Technology Experts, System-Integrator, Service-Provider; they may be from Government / Industry / Academia / Community / Consortia / R&D Institute.
System-Integrator	Person / Agency who integrates various e-Governance systems developed by the System-developer and services from Service-Providers; they may avail the services from Product-Partner, Technology Experts; they may be from Government / Industry.
Service-Provider	Person / Agency who offers e-Services and Infrastructure-Services; their services are availed by the System-developers and System-integrators; they may be from Government / Industry / Academia / Community / Consortia / R&D Institute.
Product-Partner	Person / Agency who offers product specific solution; they are registered partners from Industry / Academia / Community / Consortia / R&D Institute; they offer source-code level enhancements on the identified OSS solution.

Technology Experts	Registered Domain experts from Community, Academia, R&D Institutes and Government
End-User	Person / Agency who avails the e-Services of the system developed; they may be Citizen, Business-Organisation, Employee of a Government organisation, another Government unit.
Community	A complete ecosystem of a particular OSS solution which includes Developer, User, Partner Company and Trust.
Developer	Person who builds up the OSS solution; some are paid by the User-Company / Trust / Partner-Company / Other-Institutions; others work on a voluntary basis.
User	Person, who adopts the particular OSS solution, provides feedback and suggestions on new features, tests existing features, and offers ideas for the direction of OSS solution; some users engage the commercial support services on the OSS solution from the Partner-Companies / Trust / Developer.
Partner-Company	Organisation which offers commercial support services (like support, maintenance, training, certification, consulting, installation, enhancements & bug-fixes) on the OSS solution; receives payment (like annual fees, subscription fees, royalties) from the User and paid / unpaid works from the Developer.
Trust	A core foundation or a company that maintains and coordinates the entire project of the OSS solution; it receive annual fees from Partner-Companies; it also receives fees from the User for new features in the OSS solution.
Consultant	Person who advises Government on various e-Governance systems. The person may be from Government / Industry.

The Governments worldwide are making move towards adoption of open source software. UNDP has taken many initiatives for promotion of OSS and bringing many important reports / guidelines on OSS. Adoption of Open Source Software is easier said than done. The Government departments face difficulties in selecting the right kind of technologies commensurate to their needs. This framework would assist the Government departments in identifying and selecting the open source software as per their requirements. This framework would be helpful for e-Governance experts and practitioners, who are interested in the implementation of various open source software for different functional and technical domains.

Policy on Open Application Programming Interfaces (APIs) for Government of India

Policy onOpenApplicationProgrammingInterfaces (APIs) forGovernmentofIndia

As a part of Digital India, G2C, G2B and G2G services are to be delivered and made accessible through multiple channels like web, mobile and common service delivery outlets. Interoperability among various e-Governance applications and databases is vital for integrated service delivery. The world-wide initiatives on “Open Government” also focus on open APIs to easily access the information collected by Government organizations.

The Policy on Open APIs for Government of India sets out the Government’s approach on the use of Open APIs to promote software interoperability for all e-Governance applications & systems and provide access to data & services for promoting participation of all stakeholders including citizens.

The policy provides details on the following:

- Objectives
- Nature of compliances
- Applicability
- Implementation mechanism

Chapter 4: Policy on Open Application Programming Interfaces (APIs) for Government of India

4.1 Preamble

Under the overarching vision of Digital India, Government of India (GoI) aims to make all Government services digitally accessible to citizens through multiple channels, such as web, mobile and common service delivery outlets. To meet this objective, there is a need for an interoperable ecosystem of data, applications and processes which will make the right information available to the right user at the right time.

In order to make rapid progress in this direction, Government of India (GoI) has taken various policy initiatives, including implementation of Mission Mode Projects (MMPs). Interoperability among various e-Governance systems is an important prerequisite for upgrading the quality and effectiveness of service delivery. It is also required in order to facilitate the single window concept of electronic services delivery by Government organizations.

For promoting Open Standards for software interoperability across various Government departments and agencies, GoI has already notified the"Policy on Open Standards for e-Governance" and "Technical Standards on Interoperability Framework for e-Governance". The world-wide initiatives on "Open Government" also focus on open APIs to easily access the information collected by Government organizations.

Given the enormous advantages in this regard, there is a need to formulate a policy for the Government organizations in India to provide Open Application Programming Interfaces (APIs). The "Policy on Open APIs for Government of India" (hereinafter referred to as the "Policy") will encourage the formal use of Open APIs in Government organizations. This policy sets out the Government's approach on the use of "Open APIs" to promote software interoperability for all e-Governance applications & systems and provide access to data & services for promoting participation of all stakeholders including citizens.

What is this policy?

The Policy on Open APIs for Government of India sets out the Government's approach on the use of Open APIs to promote software interoperability for all e-Governance applications & systems and provide access to data & services for promoting participation of all stakeholders including citizens.

4.2 Objectives of the policy:

The objectives of this policy are to:

- Ensure that APIs are published by all Government organisations for all e-Governance applications and systems.
- Enable quick and transparent integration with other e-Governance applications and systems.
- Enable safe and reliable sharing of information and data across various e-Governance applications and systems.
- Promote and expedite innovation through the availability of data from e-Governance applications and systems to the public.
- Provide guidance to Government organizations in developing, publishing and implementation using these Open APIs.

4.3 Definitions

Please refer Appendix – I.

4.4 Policy Statement

Government of India shall adopt Open APIs to enable quick and transparent integration with other e-Governance applications and systems implemented by various Government organizations, thereby providing access to data & services and promoting citizen participation for the benefit of the community. The Open APIs shall have the following characteristics for publishing and consumption:

1. The relevant information being provided by all Government organisations through their respective e-Governance applications shall be open and machine readable.
2. All the relevant information and data of a Government organisation shall be made available by Open APIs, as per the classification given in the National Data Sharing and Accessibility Policy (NDSAP-2012), so that the public can access information and data.
3. All Open APIs built and data provided, shall adhere to National Cyber Security Policy.

4. The Government organizations shall make sure that the Open APIs are stable and scalable.
5. All the relevant information, data and functionalities within an e-Governance application or system of a Government organisation shall be made available to other e-Governance applications and systems through Open APIs which should be platform and language independent.
6. A Government organisation consuming the data and information from other e-Governance applications and systems using Open APIs shall undertake information handling, authentication and authorisation through a process as defined by the API publishing Organisation.
7. Each published API of a Government organization shall be provided free of charge whenever possible to other Government organizations and public.
8. Each published API shall be properly documented with sample code and sufficient information for developers to make use of the API.
9. The life-cycle of the Open API shall be made available by the API publishing Government organisation. The API shall be backward compatible with at least two earlier versions.
10. All Open API systems built and data provided shall adhere to GoI security policies and guidelines.
11. Government organizations may use an authentication mechanism to enable service interoperability and single sign-on.

4.5 Nature of Compliance

Mandatory

4.6 Applicability

The policy shall be applicable to all Government organisations under the Central Government and those State Governments that choose to adopt this policy for the following categories of e-Governance systems:

- All new e-Governance applications and systems being considered for implementation.
- New versions of the legacy and existing systems.

4.7 Implementation Mechanism

1. GoI shall formulate detailed implementation guidelines for rapid and effective adoption of the policy.
2. Government organisations shall publish the APIs so that the public can access relevant information and data from e-Governance applications and systems.

3. Government organisations shall publish the APIs for integrating with their e-Governance applications and systems.
4. Government organisations shall integrate with the e-Governance applications and systems of other departments through the messaging gateway built on open standards by the Department of Electronics and Information Technology (DeitY).
5. Government organisations shall ensure compliance with notified GoI standards for developing APIs.
6. GoI shall constitute an Implementation Committeefor facilitating the implementation of this policy and its provisions thereof.
7. GoI shall establish suitable support mechanism to facilitate API management.
8. All Government organizations, while implementing e-Governance applications and systems, must include a specific requirement in the Request for Proposal (RFP) to publish the APIs to public and other Government organizations.

4.8 Review of the Policy

GoI shall have the right to review and revise the policy as and when required.

4.9 Point of Contact

All queries or comments related to the "Policy on Open APIs for Government of India" shall be directed to the Joint Secretary (e-Governance), DeitY at jsegov@deity.gov.in.

Why we need it?

- Government organizations can offer integrated service delivery
- Published API of a Government organization shall be provided free of charge whenever possible to other Government organizations and public
- Public can access information and data
- Private players can offer innovative solutions based on data exchange using open APIs

How will it be implemented?

- Detailed implementation guidelines shall be formulated for effective adoption of the policy
- Government organisations shall publish the APIs
- An Implementation Committee shall be constituted
- Support mechanism shall be established to facilitate API management
- Request for Proposal shallhave a specific requirement to publish the

4.10 Appendix – I

Definitions

1. **API:**The term Application Programming Interface (API) means any mechanism that allows a system or service to access data or functionality provided by another system or service. The API is generally used to interact (like query, list, search, sometimes submit & update) directly with the specific information on a system, to trigger some action on other systems, or to perform some other action on other systems.
2. **Domain:** A sub-category under an Information Technology field is a Domain; specific purpose within a "Domain" is known as "Area". For example, "Document type for Web publishing content" is one Area under the "Presentation" domain.
3. **Government organization:** For the purpose of this policy, a Government organisation refers to all Ministries/ Departments/ offices/ statutory bodies/ autonomous bodies, both at the Central and State levels. Government organizations offering commercial services are not included.
4. **e-Governance:** A procedural approach in which the Government and the citizens, businesses, and other stakeholders are able to transact all or part of their activities using Information and Communication Technology tools.
5. **Systems:** A group of interacting, interrelated, or interdependent elements forming a complex whole. Information System is a combination of people, hardware, software, communication devices, network and data resources that processes (can be storing, retrieving, transforming information) data and information for a specific purpose.

6. **Legacy System:** An old method, technology, computer system, or application program that continues to be used, typically because it still functions for the users' needs, even though newer technology or more efficient methods of performing a task are now available.
7. **New version of Legacy System:** The legacy system which has undergone a major version change due to re-engineering like functional changes, architectural changes, technology changes, change in storage mechanism, design implementation changes etc.
8. **Open API:** Open API is the API that has been exposed to enable other systems to interact with that system. Open API may be either integrated with the host application or may be an additional piece of software that exposes any proprietary API with an Open API equivalent. The Open API, whenever possible, may be free of charge and without restrictions for reuse & modifications.
9. **Policy on Open Standards for e-Governance:**The Policy on Open Standards for e-Governance provides a framework for the selection of Standards to facilitate interoperability between systems developed by multiple agencies. It is available at https://egovstandards.gov.in/sites/default/files/Policy/Policy%20On%20Open%20Standards/Policy_on_Open_Standards_for_e-Governance_Ver1.0.pdf
10. **Technical Standards on Interoperability Framework for e-Governance:**This document describes technical standards to be adopted for e-Governance application in the areas covered, as per the Policy on Open Standards for e-Governance. Available at https://egovstandards.gov.in/sites/default/files/Published_Standards/Technical%20Standards%20for%20IFEG/Technical_Standards_for_IFEG_Ver1.0.pdf.
11. **National Data Sharing and Accessibility Policy (NDSAP-2012):**The objective of this policy is to facilitate access to Government of India owned shareable data and information in both human readable and machine readable forms through a network all over the country in a proactive and periodically updatable manner, within the framework of various related policies, Acts and rules of Government of India, thereby

permitting a wider accessibility and use of public data and information. It is available at http://ogpl.gov.in/NDSAP/NDSAP-30Jan2012.pdf

12. **National Cyber Security Policy 2013:** The objective of this policy is to protect information and information infrastructure in cyberspace, build capabilities to prevent and respond to cyber threats, reduce vulnerabilities and minimize damage from cyber incidents through a combination of institutional structures, people, processes, technology and cooperation. It is available at http://deity.gov.in/content/national-cyber-security-policy-2013-1

Though there are a number of e-Governance applications, each one delivering some services, citizens are usually required to contact individual departments whenever they need services concerning multiple departments. This is because the departmental applications/ databases are not inter-connected and do not have data exchange facility amongst themselves. This creates hassles for common persons as they are not able to get end-to-end service through a single window mechanism.

With the adoption of open APIs and integration amongst applications through open APIs, citizens would be able to get various services by filling a single application form, even when such an integrated service might need processing from multiple Government departments and agencies. In the background, systems would talk to each other and would facilitate data and information exchange leading to service delivery at the end. This would provide convenience to both citizens and businesses in availing various Government services.

For example, various travel sites are currently providing the railway ticketing services as their systems can talk to railway systems in real time, which is being facilitated through open APIs. Once the e-Governance projects start adopting open APIs, common citizens will be able to avail multiple services by submitting single application form.

Email Policy of Government of India

E-mail Policy of Government of India

This policy lays down the guidelines with respect to use of e-mail services. It is applicable to all employees of GoI and employees of those State/UT Governments that use the e-mail services of GoI and also those State/UT Governments that choose to adopt this policy in future.The objective of this policy is to ensure secure access and usage of Government of India e-mail services by its users.

The policy covers the following aspects:

- Security
- E-mail Account Management
- Delegated Admin Console
- E-mail Domain & Virtual Hosting
- Use of Secure Passwords
- Privacy
- Responsibilities of User Organizations
- Responsibilities of Users
- Service Level Agreement
- Scrutiny of e-mails/Release of logs
- Security Incident Management Process
- Intellectual Property
- Enforcement
- Deactivation
- Exemption
- Audit

Chapter 5: Email Policyof Govt. of India

5.1 Introduction

The Government uses e-mail as a major mode of communication. Communications include Government of India (GoI) data that travel as part of mail transactions between userslocated both within the country and outside.

This policy of Government of India lays down the guidelines with respect to use of e-mail services. The Implementing Agency (IA) for the GoI e-mail service shall be National Informatics Centre (NIC), under the Department of Electronics and Information Technology (DeitY), Ministry of Communications and Information Technology. The organisations exempted under Clause 14 will themselves become the Implementing Agency (IA) for the purpose of this policy.

5.2 Scope

- Only the e-mail services provided by NIC, the Implementing Agency of the Government of India shall be used for official communications by all organizations except those exempted under clause no 14 of this policy. The e-mail services provided by other service providers shall not be used for any official communication.
- This policy is applicable to all employees of GoI and employees of those State/UT Governments that use the e-mail services of GoI and also those State/UT Governments that choose to adopt this policy in future. The directives contained in this policy must be followed by all of them with no exceptions. All users of e-mail services can find further information in the supporting policies available on http://www.deity.gov.in/content/policiesguidelines under the caption "E-mail Policy".
- E-mail can be used as part of the electronic file processing in Government of India. Further information in this regard is available at: http://darpg.gov.in/darpgwebsite_cms/Document/file/CSMeOP_1st_Edition.pdf.

5.3 Objectives

- The objective of this policy is to ensure secure access and usage of Government of India e-mail services by its users. Users have the responsibility to use this resource in an efficient, effective, lawful, and ethical manner. Use of the Government of India e-mail service amounts to the user's agreement to be governed by this policy.

- All services under e-mail are offered free of cost to all officials under Ministries / Departments / Statutory Bodies / Autonomous bodies (henceforth referred to as "Organization" in the policy) of both Central and State/UT Governments. More information is available under "NIC e-mail Services and Usage Policy" at http://www.deity.gov.in/content/policiesguidelines/ under the caption "E-mail Policy".
- Any other policies, guidelines or instructions on e-mail previously issued shall be superseded by this policy.

5.4 Roles specified for implementation of the Policy

The following roles are specified in each organization using the GoI e-mail service. The official identified for the task shall be responsible for the management of the entire user base configured under that respective domain.

- Competent Authority as identified by each organization
- Designated nodal officer as identified by each organization
- GoI e-mail service Implementing Agency (IA), i.e. National Informatics Centre or the exempt organisation as per Clause 14 of this policy

What is this policy?

- Provides policy guidelines with respect to use of e-mail services
- It is applicable to all employees of GoI and employees of those State/UT Governments that use the e-mail services of GoI and also those State/UT Governments that choose to adopt this policy in future

5.5Basic Requirements of GoI e-mail Service

5.5.1 Security

(a) Considering the security concerns with regard to a sensitive deployment like e-mail, apart from the service provided by the IA, there would not be any other e-mail service under GoI.

(b) All organizations, except those exempted under clause 14 of this policy, should migrate their e-mail

(c) Deployment of the IA for security reasons and uniform policy enforcement. For the purpose of continuity, the e-mail address of the organization migrating their service to the IA deployment shall be retained as part of the migration process. Wherever it is technically feasible, data migration shall also be done.

(d) Secure access to the GoI email service

a. It is recommended for users working in sensitive offices to use VPN/OTP for secure authentication as deemed appropriate by the competent authority.

b. It is recommended that GoI officials on long deputation/stationed abroad and handling sensitive information should use (VPN)/ (OTP)for accessing GoI e-mail services as deemed appropriate by the competent authority.

c. It is recommended that Embassies and missions abroad should use Static IP addresses for accessing the services of the IAas deemed appropriate by the competent authority.

d. More information is available under "Guidelines for E-mail Management and Effective E-mail Usage" at services to the centralized http://www.deity.gov.in/content/policiesguidelines under the caption "E-mail Policy".

(e) From the perspective of security, the following shall be adhered to by all users of GoI e-mail service:

a. Relevant Policies framed by Ministry of Home Affairs, relating to classification, handling and security of information shall be followed.

b. Use of Digital Signature Certificate (DSC) and encryption shall, be mandatory for sending e-mails deemed as classified and sensitive, in accordance with the relevant policies of Ministry of Home Affairs.Updation of current mobile numbers under the personal profile of users is mandatory for security reasons. The number would be used only for alerts and information regarding security sent by the IA. Updation of personal e-mail id (preferably from a service provider within India),in addition to the mobile number, shall also be mandatory in order to reach the user through an alternate means for sending alerts.

c. Users shall not download e-mails from their official e-mail account, configured on the GoI mail server, by configuring POPor IMAP on any other e-mail service provider. This implies that users should not provide their GoI e-mail account details (id and password) to their accounts on private e-mail service providers.

d. Any e-mail addressed to a user, whose account has been deactivated /deleted, shall not be redirected to another e-mail address. Such e-mails may contain contents that belong to the Government and hence no e-mails shall be redirected.

e. The concerned nodal officer of the organization shall ensure that the latest operating system, anti-virus and application patches are available on all the devices, in coordination with the User.

f. In case a compromise of an e-mail id is detected by the IA, an SMS alert shall be sent to the user on the registered mobile number. In case an “attempt” to compromise the password of an account is detected, an e-mail alert shall be sent. Both the e-mail and the SMS shall contain details of the action to be taken by the user. In case a user does not take the required action even after five such alerts (indicating a compromise), the IA reserves the right to reset the password of that particular e-mail id under intimation to the nodal officer of that respective organization.

g. In case of a situation when a compromise of a user id impacts a large user base or the data security of the deployment, the IA shall reset the password of that user id. This action shall be taken on an immediate basis, and the information shall be provided to the user and the nodal officer subsequently. SMS shall be one of the prime channels to contact a user; hence all users should ensure that their mobile numbers are updated.

h. Forwarding of e-mail from the e-mail id provided by GoI to the Government official’s personal id outside the GoI e-mail service is not allowed due to security reasons. Official e-mail id provided by the IA can be used to communicate with any other user, whether private or public. However, the user must exercise due discretion on the contents that are being sent as part of the e-mail.

i. Auto-save of password in the Government e-mail service shall not be permitted due to security reasons.

j. More details regarding security measures are available in “NIC Security Policy” at http://www.deity.gov.in/content/policiesguidelines under the caption “E-mail Policy”.

k. The guidelines for effective e-mail usage have been described in “Guidelines for E-mail Account Management and Effective E-mail Usage”available at http://www.deity.gov.in/content/policiesguidelines under the caption “Email Policy”.

Why we need it?

- To ensure secure access and usage of Government of India e-mail services by its users
- To ensure use of e-mail service in an efficient, lawful and ethical manner

5.5.2 E-mail Account Management

(a) Based on the request of the respective organizations, IA will create two ids, one based on the designation and the other based on the name. Designation based id’s are recommended for officers dealing with the public. Use of alphanumeric characters as part of the e-mail id is recommended for sensitive users as deemed appropriate by the competent authority.

(b) Government officers who resign or superannuate after rendering at least 20 years of service shall be allowed to retain the name based e-mail address i.e. userid@gov.in for one year post resignation or superannuation. Subsequently, a new e-mail address with the same user id but with a different domain address (for instance,userid@pension.gov.in), would be provided by the IA for their entire life.

More details pertaining to e-mail account management are provided in “Guidelines for E-mail Account Management and Effective E-mail Usage” availableat http://www.deity.gov.in/content/policiesguidelines under the caption “Email Policy”. The document covers creation of E-mail addresses, process of account creation, process of handover of designation-based ids, status of account after resignation and superannuation, data retention & backup and deactivation of accounts.

5.5.3 Delegated Admin Console

Organizations can avail the “Delegated Admin Console" service from IA. Using the console the authorized person of an organization can create/delete/change the password of user ids under that respective domain as and when required without routing the request through IA. Organizations that do notopt for the admin console need to forward their requests with complete details to the IA's support cell (support@gov.in).

5.5.4 E-mail Domain & Virtual Hosting

a) GoI provides virtual domain hosting for e-mail. If an organization so desires, the IA can offer a domain of e-mail addresses as required by them. This implies that if an organization requires an address resembling the website that they are operating, IA can provide the same.

b) By default, the address "userid@gov.in" shall be assigned to the users. The user id shall be created as per the addressing policy available at http://www.deity.gov.in/content/policiesguidelines/ under "E-mail Policy".

C) Organizations desirous of an e-mail address belonging to other domains (e.g. xxxx@deity.gov.in, yyyy@tourism.gov.in) need to forward their requests to the IA

5.5.5 Use of Secure Passwords

All users accessing the e-mail services must use strong passwords for security of their e-mail accounts. More details about the password policy are available in "Password Policy" at http://www.deity.gov.in/content/policiesguidelines under the caption "E-mail Policy".

How will it be implemented?

- Each organization shall be responsible for ensuring compliance with the provisions of this policy. The Implementing Agency would provide the necessary technical assistance to the organizations in this regard.Detailed implementation guidelines shall be formulated for rapid and effective adoption of the policy.

5.5.6 Privacy

Users should ensure that e-mails are kept confidential. IA shall take all possible precautions on maintaining privacy. Users must ensure that information regarding their password or any other personal information is not shared with anyone.

5.6 Responsibilities of User Organizations

5.6.1 Policy Compliance

a) All user organizations shall implement appropriate controls to ensure compliance with the e-mail policy by their users. IA shall give the requisite support in this regard.

b) The user organizations shall ensure that official e-mail accounts of all its users are created only on the e-mail server of the IA**.**

c) Nodal officer of the user organization shall ensure resolution of all incidents related to the security aspects of the e-mail policy. IA shall give the requisite support in this regard.

d) Competent Authority of the user organization shall ensure that training and awareness programs on e-mail security are organized at regular intervals. Implementing Agency shall provide the required support.

5.6.2 Policy Dissemination

a) Competent Authority of the concerned organization should ensure dissemination of the e-mail policy.

b) Competent Authority should use Newsletters, banners, bulletin boards etc, to facilitate increased awareness on the e-mail policy.

c) Orientation programs for new recruits shall include a session on the e-mail policy.

5.7 Responsibilities of Users

5.7.1 Appropriate Use of e-mail Service

(a) E-mail is provided as a professional resource to assist users in fulfilling their official duties. Designation based ids should be used for official communication and name based ids can be used for both official and personal communication.

(b) **Examples of inappropriate use of the e-mail service**

a. Creation and exchange of e-mails that could be categorized as harassing, obscene or threatening.

b. Unauthorized exchange of proprietary information or any other privileged, confidential or sensitive information.

c. Unauthorized access of the services. This includes the distribution of e-mails anonymously, use of other officers' user ids or using a false identity.

d. Creation and exchange of advertisements, solicitations, chain letters and other unofficial, unsolicited e-mail.

e. Creation and exchange of information in violation of any laws, including copyright laws.

f. Wilful transmission of an e-mail containing a computer virus.

g. Misrepresentation of the identity of the sender of an e-mail.

h. Use or attempt to use the accounts of others without their permission.

i. Transmission of e-mails involving language derogatory to religion, caste, ethnicity, sending personal e-mails to a broadcast list, exchange of e-mails containing anti-national messages, sending e-mails with obscene material, etc.

j. Use of distribution lists for the purpose of sending e-mails that are personal in nature, such as personal functions, etc.

Any case of inappropriate use of e-mail accounts shall be considered a violation of the policy and may result in deactivation of the account. Further, such instances may also invite scrutiny by the investigating agencies depending on the nature of violation.

5.7.2 User's Role

(a) The User is responsible for any data/e-mail that is transmitted using the GoI e-mail system. All e-mails/data sent through the mail server are the sole responsibility of the user owning the account.

(b) Sharing of passwords is prohibited.

(c) The user's responsibility shall extend to the following:

a. Users shall be responsible for the activities carried out on their client systems, using the accounts assigned to them.

b. The 'reply all' and the use of 'distribution lists' should be used with caution to reduce the risk of sending e-mails to wrong people.

c. Back up of important files shall be taken by the user at regular intervals. The IA shall not restore the data lost due to user's actions.

5.8 Service Level Agreement

The IA shall provide the e-mail services based on the Service Level Agreement (SLA) available at http://www.deity.gov.in/content/policiesguidelines under the caption "E-mail Policy".

5.9 Scrutiny of e-mails/Release of logs

1. Notwithstanding anything in the clauses above, the disclosure of logs/e-mails to law enforcement agencies and other organizations by the IA would be done only as per the

IT Act 2000 and other applicable laws.

2. The IA shall neither accept nor act on the request from any other organization, save as provided in this clause, for scrutiny of e-mails or release of logs.
3. IA will maintain logs for a period of two years.

5.10 Security Incident Management Process

1. A security incident is defined as any adverse event that can impact the availability, integrity, confidentiality and authority of Government data. Security incidents can be due to factors like malware, phishing, loss of a device, compromise of an e-mail id etc.
2. It shall be within the right of the IA to deactivate or remove any feature of the e-mail service if it is deemed as a threat and can lead to a compromise of the service.
3. Any security incident, noticed or identified by a user must immediately be brought to the notice of the Indian Computer Emergency Response Team (ICERT) and the IA.

5.11Intellectual Property

Material accessible through the IA's e-mail service and resources may be subject to protection under privacy, publicity, or other personal rights and intellectual property rights, including but not limited to, copyrights and laws protecting patents, trademarks, trade secrets or other proprietary information. Users shall not use the Government service and resources in any manner that would infringe, dilute, misappropriate, or otherwise violate any such rights.

5.12Enforcement

1. This “E-mail policy” is applicable to all Government employees as specified in clause 2.2.
2. Each organization shall be responsible for ensuring compliance with the provisions of this policy. The Implementing Agency would provide necessary technical assistance to the organizations in this regard.

5.13Deactivation

1. In case of threat to the security of the Government service, the e-mail id being used to impact the service may be suspended or deactivated immediately by the IA.
2. Subsequent to deactivation, the concerned user and the competent authority of that respective organization shall be informed.

5.14 Exemption

1. Organizations, including those dealing with national security, that currently have their own independent mail servers can continue to operate the same, provided the e-mail servers are hosted in India. These organizations however need to ensure that the principles of the e-mail policy are followed. However, in the interest of uniform policy enforcement and security, it is recommended that these organizations should consider migrating to the core service of the IA.
2. Indian Missions and Posts abroad may, however, maintain alternative e-mail services hosted outside India to ensure availability of local communication channels under exigent circumstances such as disruption of internet services that can cause non-availability of Government e-mail services.
3. Organizations operating Intranet [13] mail servers with air-gap are exempted from this policy.

5.15 Audit of E-mail Services

The security audit of NIC email services and other organizations maintaining their own mail server shall be conducted periodically by an organization approved by Deity.

5.16 Review

Future changes in this Policy, as deemed necessary, shall be made by DeitY with approval of the Minister of Communication & IT after due inter-ministerial consultations.

5.17 Glossary

S.N.	TERM	DEFINITION
1	**Users**	Refers to Government/State/UT employees who are accessing the Government e-mail services.
2	**Implementing agency (IA)**	For the purpose of this policy, the implementing agency is "National Informatics Centre" under the Department of Electronics and Information Technology, Ministry of Communications and Information Technology, Government of India
3	**Organization**	For the purpose of this policy, organisation refers to all ministries/departments/offices/statutory bodies/autonomous bodies, both at the Central and State level. Government organizations offering commercial services are not included.
4	**Competent Authority**	Officer responsible for taking and approving all decisions relating to this policy in his Organization
5.	**Nodal Officer**	Officer responsible for all matters relating to this policy who will coordinate on behalf of the Organization
6	**DSC**	A **digital signature** is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives the recipient reason to believe that the e-mail was created by a known sender, such that the sender cannot deny having sent the e-mail (authentication and non-repudiation) and that the e-mail was not altered in transit (integrity).
7	**VPN**	A **virtual private network** extends a private network across a public network, such as the Internet. It enables a computer to send and receive data across shared or public networks as if it were directly connected to the private network, while benefitting from the functionality, security and management policies of the private network
8	**OTP**	A **one-time password** (OTP) is a password that is valid for only one login session or transaction. OTPs avoid a number of shortcomings that are associated with traditional (static) passwords.

9	**POP**	**POP i**s short for **P***ost* **O***ffice* **P***rotocol*, a protocol used to retrieve e-mail from a mail server.
10	**IMAP**	IMAP is short for **“The Internet Message Access Protocol”,** a protocol used to retrieve e-mail from a remote mail server. Unlike POP, in IMAP, Messages are displayed on your local computer but are kept and stored on the mail server. IMAP allows you to sync your folders with the e-mail server which is not possible using POP.
11	**Deactivation**	**Deactivation** of an account means that the account can no longer be accessed. All e-mails sent to a deactivated account shall bounce to the sender
12	**Phishing**	**Phishing** is a fraudulent attempt, usually made through e-mail, to steal a user’s personal information. Phishing e-mails almost always tell a user to click a link that takes the user to a site from where the personal information is requested. Legitimate organisations would never request this information via e-mail. Users should never click on a link. A user should always type a URL in the browser even if the link appears genuine**.**
13	**Intranet**	An intranet is a private network that is contained within an organization. For the purpose of this policy, computers connected to an intranet are not allowed to connect to internet.

E-mail has become a major mode of information exchange amongst Government officials. With the computerization and automation of Government offices, the usage of email would keep on increasing in the future. This policy provides guidelines with respect to secure access and usage of the e-mail services provided by the Implementing Agency of the Government of India for official communications by all organizations except those that aregranted exemption. This provides an assurance that official communication and data are safe and secure.

Policy on Use of IT Resources

Policy on Use of IT Resources of Government of India

This policy governs the usage of IT resources from an end user's perspective**.** The objective of this policy is to ensure proper access to and usage of Government's IT resources and prevent their misuse by the users. This policy is applicable to all employees of GoI and employees of those State/UT Governments that use the IT resources of GoI and also those State/UT Governments that choose to adopt this policy in future.

The policy provides details on the following:

- Scope
- Objectives
- Roles and responsibilities
- Access to the network
- Monitoring and Privacy
- Access to social media sites
- Security Incident Management Process
- Intellectual Property
- Enforcement
- Deactivation
- Audit

Chapter 6: Policy on Use of IT Resources of Government of India

6.1 Introduction

Government provides IT resources to its employees to enhance their efficiency and productivity. These resources are meant as tools to access and process information related to their areas of work. These resources help Government officials to remain well informed and carry out their functions in an efficient and effective manner.

For the purpose of this policy, the term 'IT Resources' includes desktop devices, portable and mobile devices, networks including wireless networks, Internet connectivity, external storage devices and peripherals like printers and scanners and the software associated therewith.

Misuse of these resources can result in unwanted risk and liabilities for the Government. It is, therefore, expected that these resources are used primarily for Government related purposes and in a lawful and ethical way.

6.2 Scope

This policy governs the usage of IT Resources from an end user's perspective.This policy is applicable to all employees of GoI and employees of those State/UT Governments that use the IT Resources of GoI and also those State/UT Governments that choose to adopt this policy in future

What is this policy?

- Governs the usage of IT resources from an end user's perspective
- Applicable to all employees of GoI and employees of those State/UT Governments that use the IT resources of GoI and also those State/UT Governments that choose to adopt their policy in future

6.3 Objective

The objective of this policy is to ensure proper access to and usage of Government's IT resources and prevent their misuse by the users. Use of resources provided by Government of India implies the user's agreement to be governed by this policy.

6.4 Roles and Responsibilities

The following roles are required in each organization using the Central / State / UT Government IT resources.

The official identified for the task shall be responsible for the management of the IT resources deployed for the use of entire user base under their respective domain.

(1) Competent Authority as identified by each organization.

(2) Designated Nodal Officer as identified by each organization.

(3) Implementing Agency: The overall responsibility for Information Security will be that of the respective organization. In the interest of security of the network services, it is recommended that the organizations should use the GoI network services provided by NIC, in which case NIC would be the Implementing Agency for security of network services on behalf of the concerned organization. In organizations not using NIC network services, the respective organization will be the Implementing Agency.

(4) The Nodal Agency for managing all IT Resources except network services shall be the respective organization.

6.5 Access to the Network

6.5.1 Access to Internet and Intranet

a. A user shall register the client system and obtain one time approval from the competent authority before connecting the client system to the Government network.

b. It is strongly recommended that sensitive offices shall maintain two independent networks, i.e. Internet and Intranet. Both the networks shall not have any physical connection/devices between them. Users in such deployments shall have two access devices, i.e. desktops. One shall be connected to the internet and the other to the intranet. End point compliance shall be implemented on both the networks to prevent unauthorised access to data.

c. Users shall not undertake any activity through any website or applications to bypass filtering of the network or perform any other unlawful acts which may harm the network's performance or security

6.5.2 Access to Government Wireless Networks

For connecting to a Government wireless network, user shall ensure the following:

a. A user shall register the access device and obtain one time approval from the competent authority before connecting the access device to the Government wireless network.

b. Wireless client systems and wireless devices shall not be allowed to connect to the Government wireless access points without due authentication.

Why we need it?

Ensure proper access to and usage of Government's IT resources and prevent their misuse by users

c. To ensure information security, it is recommended that users should not connect their devices to unsecured wireless networks.

6.5.3 Filtering and blocking of sites:

a. IA may block content over the Internet which is in contravention of the relevant provisions of the IT Act 2000 and other applicable laws or which may pose a security threat to the network.

b. IA may also block content which, in the opinion of the organization concerned, is inappropriate or may adversely affect the productivity of the users.

6.6 Monitoring and Privacy:

(1) IA shall have the right to audit networks and systems at regular intervals, from the point of compliance to this policy.

(2) IA/Nodal Agency, for security related reasons or for compliance with applicable laws, may access, review, copy or delete any kind of electronic communication or files stored on Government provided devices under intimation to the user. This includes items such as files, e-mails, and Internet history etc.

(3) IA may monitor user's online activities on Government network, subject to such Standard Operating Procedures as the organization may lay down in this regard.

6.7 E-mail Access from the Government Network

(1) Users shall refrain from using private e-mail servers from Government network.

(2) E-mail service authorized by the Government and implemented by the IA shall only be used for all official correspondence. For personal correspondence, users may use the name-based e-mail id assigned to them on the Government authorized e-mail Service.

(3) More details in this regard are provided in the “E-mail Policy of Government of India”.

6.8 Access to Social Media Sites from Government Network

(1) Use of social networking sites by Government organizations is governed by “Framework and Guidelines for use of Social Media for Government Organizations” available at http://deity.gov.in.

(2) User shall comply with all the applicable provisions under the IT Act 2000, while posting any data pertaining to the Government on social networking sites.

(3) User shall adhere to the “Terms of Use” of the relevant social media platform/website, as well as copyright, privacy, defamation, contempt of court, discrimination, harassment and other applicable laws.

(4) User shall report any suspicious incident as soon as possible to the competent authority.

(5) User shall always use high security settings on social networking sites.

(6) User shall not post any material that is offensive, threatening, obscene, infringes copyright, defamatory, hateful, harassing, bullying, discriminatory, racist, sexist, or is otherwise unlawful.

(7) User shall not disclose or use any confidential information obtained in their capacity as an employee/contractor of the organization.

(8) User shall not make any comment or post any material that might otherwise cause damage to the organization’s reputation.

6.9 Use of IT Devices Issued by Government of India

IT devices issued by the Government to a user shall be primarily used for Government related purposes and in a lawful and ethical way and shall be governed by the practices defined in the document **“Guidelines for Use of IT Devices on Government Network”** available at http://www.deity.gov.in/content/policiesguidelines/ under the caption “Policy on Use of IT Resources”. The aforesaid document covers best practices related to use of desktop devices, portable devices, external storage media and peripherals devices such as printers and scanners.

6.10 Responsibility of User Organizations

6.10.1 Policy Compliance

a. All user organizations shall implement appropriate controls to ensure compliance with this policy by their users. Implementing Agency shall provide necessary support in this regard.

b. A periodic reporting mechanism to ensure the compliance of this policy shall be established by the competent authority of the organization.

c. Nodal Officer of the user organization shall ensure resolution of all incidents related to the security aspects of this policy by their users. Implementing Agency shall provide the requisite support in this regard.

d. Competent Authority of the user organization shall ensure that training and awareness programs on use of IT resources are organized at regular intervals. Implementing Agency shall provide the required support in this regard.

e. User organization shall not install any network/security device on the network without consultation with the IA.

How will it be implemented?

- Official identified for the task shall be responsible for the management of the IT resources deployed for use under their respective domain
- Overall responsibility for Information Security will be that of the respective organization
- It is recommended that the organizations should use the GoI network services provided by NIC, in which case NIC would be the Implementing Agency for security of network services on behalf of the concerned organization
- In organizations not using NIC network services, the respective organization will be the Implementing Agency
- Network, monitoring and privacy, e-mail access, access to social media sites, use of devices, incident management, audit of logs, intellectual property, enforcement, deactivation are also covered

6.10.2 Policy Dissemination

a. Competent Authority of the user organization should ensure proper dissemination of this policy.

b. Competent Authority may use newsletters, banners, bulletin boards etc. to facilitate increased awareness about this policy amongst their users.

c. Orientation programs for new recruits shall include a session on this policy.

6.11 Security Incident Management Process

(1) A security incident is defined as any adverse event that can impact the availability, integrity, confidentiality and authority of Government data.

(2) IA reserves the right to deactivate/remove any device from the network if it is deemed as a threat and can lead to a compromise of a system under intimation to the competent authority of that organization.

(3) Any security incident noticed must immediately be brought to the notice of the Indian Computer Emergency Response Team (ICERT) and the IA.

6.12 Scrutiny/Release of logs

(1) Notwithstanding anything in the above clause, the disclosure of logs relating to or contained in any IT Resource, to Law Enforcement agencies and other organizations by the IA shall be done as per the IT Act 2000 and other applicable laws.

(2) IA shall neither accept nor act on the request from any other organization, save as provided in this clause, for scrutiny or release of logs.

6.13 Intellectual Property

Material accessible through the IA's network and resources may be subject to protection under privacy, publicity, or other personal rights and intellectual property rights, including but not limited to, copyrights and laws protecting patents, trademarks, trade secrets or other proprietary information. Users shall not use the Government network and resources in any manner that would infringe, dilute, misappropriate, or otherwise violate any such rights.

6.14 Enforcement

(1) This policy is applicable to all employees of Central and State Governments as specified in clause 2 of this document. It is mandatory for all users to adhere to the provisions of this policy.

(2) Each organization shall be responsible for ensuring compliance with the provisions of this policy. The Implementing Agency would provide necessary technical assistance to the organizations in this regard.

6.15 Deactivation

(1) In case of any threat to security of the Government systems or network from the resources being used by a user, the resources being used may be deactivated immediately by the IA.

(2) Subsequent to such deactivation, the concerned user and the competent authority of that organization shall be informed.

6.16Audit of NIC Network Infrastructure

The security audit of NIC network infrastructure shall be conducted periodically by an organization approved by Deity.

6.17 Review

Future changes in this Policy, as deemed necessary, shall be made by DeitY with approval of the Minister of Communication & IT after due inter-ministerial consultations.

6.18 Glossary

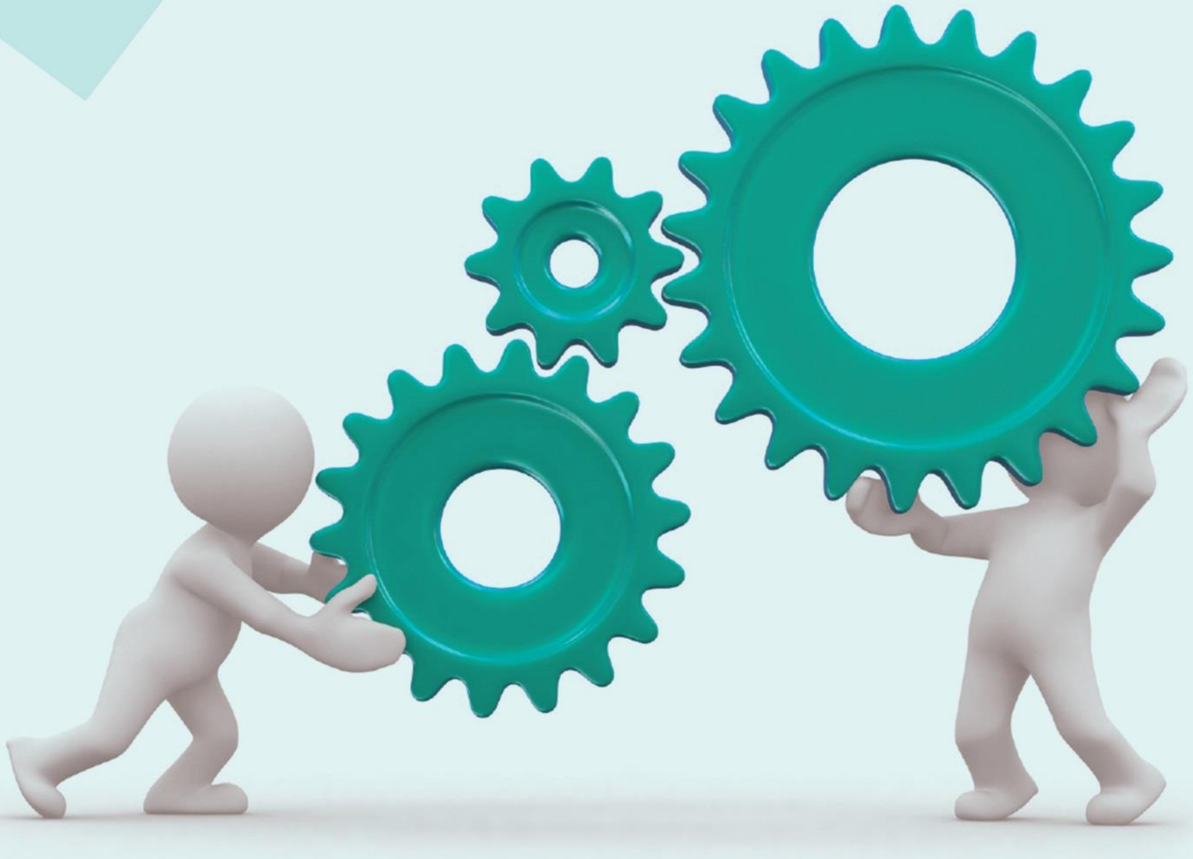
S no.	Term	Definition
1	**Users**	Refers to Government/State/UT employees/contractual employees who are accessing the Government services
2	**Organization**	Ministry/Department/Statutory Body/Autonomous body under Central and State Governments
3	**Competent Authority**	Officer responsible for taking and approving all decisions relating to this policy in his Organization
4.	**Nodal Officer**	Officer responsible for all matters relating to this policy who will coordinate on behalf of the Organization
5	**Implementing Agency (IA)**	A Body which will be responsible for ensuring compliance with this policy with reference to network services including power to take precautionary and penal actions as specified in this policy.
6	**Nodal Agency**	Respective organization responsible for ensuring compliance with this policy with respect to use of It resources except network services.

7	**Internet**	Internet is a network of the interlinked computer networking worldwide, which is accessible to the general public. These interconnected computers work by transmitting data through a special type of packet switching which is known as the IP or the internet protocol
8	**Intranet**	An intranet is a private network that is contained within an organization. For the purpose of this policy, computers connected to an intranet are not allowed to connect to internet.
9	**End point compliance**	End point compliance is an approach to network protection that requires each computing device on a network to comply with certain standards before network access is granted. Endpoints can include desktops, laptops, smart phones, tablets etc
10	**Wireless**	Any type of computer network that uses wireless data connections for connecting network nodes. For the purpose of this policy, all the GoI wireless networks will be deployed in a secure manner.
11	**Social Media**	Applies to social networking sites, blogs, electronic newsletters, online forums, social networking sites, and other services that permit users to share information with others in a contemporaneous manner.
12	**Contractor/contractual employees**	An employee who works under contract for GoI. A contract employee is hired for a specific job or assignment. A contract employee does not become a regular addition to the GoI staff and is not considered a permanent employee of GoI
13	**Security Incident**	Any adverse event which occurs on any part of the government data and results in security threat/breach of the data

Policy on Use of IT Resources

ICT resources like laptops, desktops, pen drives, networking devices, e-mail accounts, etc have become an important part of work in Government offices. The optimum use of ICT resources are helping Government departments in increasing their productivity by automating their internal processes as well as the delivery of citizen centric services. The proper use of ICT resources has become even more important due to emergence of new platforms like mobile and devices for official work.

The compliance of policy would ensure that Government’s ICT resources are used in a proper, safe and secure manner.



Policy On Collaborative Application Development by Opening the Source Code of Government Applications

Policy on Collaborative Application Development by Opening the Source Code of Government Applications

This policy aims to increase the pace of e-Governance application development and rapid roll out/implementation by adopting an open-source development model. By opening the source code, the Governments wants successful, scalable, high quality e-Governance applications to be developed in a collaborative manner. It also intends to encourage innovative e-Governance applications and solutions through collaborative development.

The policy provides details on the following:

- Background
- Objectives
- Applicability
- Responsibilities
- Review

Chapter 7: Policy on Collaborative Application Development by Opening the Source Code of Government Applications

7.1Metadata

S. N.	Data elements	Values
1.	**Title**	Policy on Collaborative Application Development by Opening the Source Code of Applications.
2.	**Document Version, Creation date**	**Version 1.0**
3.	**Publisher**	Ministry of Communication and Information Technology, Department of Electronics and Information Technology (DeitY)
4.	**Date of Publishing**	Date of Notification
5.	**Type of Standard Document** *(Policy/Technical Specification/Best Practice/Guideline/Process)*	Policy
6.	**Creator** *(An entity primarily responsible for making the resource)*	Ministry of Communications and Information Technology, Department of Electronics and Information Technology (DeitY)

S. N.	Data elements	Values
7.	**Contributor** *(An entity responsible for making contributions to the resource)*	DeitY, Govt. of India, Jharkhand IT department & NIC Jharkhand State Unit, Ranchi
8.	**Brief Description**	The policy intends to increase the pace of e-Goverance application development and rapid roll out/implementation by adopting an open-source development model. The Government of India wants to promote re-use of existing developed applications. By opening the source code, the Govt. wants successful, scalable, high quality eGov applications to be developed in a collaborative manner. It also wants new applications to be developed to encourage creativity - both inside and outside the Government by encouraging collaborative development between Govt. departments/agencies and private organizations, citizens and developers to create innovative eGov applications and solutions. eGov application source open approach including the use and release of application source code to public can reduce costs and development time and improve the overall quality and security through increased transparency and mass peer review.

S. N.	Data elements	Values
9.	**Target Audience** *(Who would be referring / using the document)*	All Central and State Government Departments, and other Government Agencies providing public services electronically, Government & private organizations engaged by Government departments, other application developers, OEMs, Audit Agencies etc
10.	**Owner of approved Policy**	Ministry of Communication and Information Technology, Department of Electronics and Information Technology (DeitY)
11.	**Coverage Spatial**	India
12.	**Language** *(To be translated in other Indian languages later)*	English (To be translated in other Indian languages later)
13.	**Copyrights**	Ministry of Communications and Information Technology, Department of Electronics and Information Technology (DeitY)
14.	**Source** **(***Reference to the resource from which present resource is derived)*	NIL
15.	**Relation** *(Relation with other e-Governance standards notified by DeitY)*	N/A

What is this policy?

It deals withcollaborative application development by opening the source code of Government applicationsto ensure a new and agile way of developing software, reuse and rapid roll out to other Government domains

7.2 Preamble

The Government of India (GOI) aims to make public the source code of various software applications/components/products as it may consider suitable and whose Intellectual Property Rights (IPR) are held by various Government entities. It is intended that this will serve the purpose of reuse, faster delivery, product standardization, innovation, quality improvement and cost saving through collaborative development.

7.3 Effective date

This policy comes in force from the date of its publication.

7.4 Background

Government Departments and Agencies both at the centre and states are engaged in developing software applications and most such applications are running successfully in their own premises. However, there may be repetitive work going on. Many applications are being re-developed from scratch without reusing the already existing and running applications in other Departments. In the absence of a common Collaborative Application Development Platform, individual applications developed by Government Departments may end up with the same code being rewritten for similar application functionality, which is already available elsewhere. Lack of sharing of the source code prevents the code from scrutiny, thus denying the opportunity for further improvements. These inefficient practices may lead to wastage of time, efforts and public money, which could have been put to more productive use alternatively.

Several hundreds of custom application software are running across central/ state Government Departments and Agencies, PSUs and urban local bodies. Hosting of the source code of these applications on a single unified platform which can be accessed by Government Departments/Agencies and the general public (with necessary access controls) would result in much faster application development in a better collaborative manner.

Why we need it?

- To ensure collaboration in application development
- To foster ecosystem of innovative solutions and application development
- For rapid replication of successful e-Governance solutions

7.5 Objectives

The "**Policy on Collaborative Application Development by Opening the Source Code of Government Applications**" is designed with the objective of promoting reuse, standardization, innovation, quality improvement and cost savings through collaboration and avoidance of duplication.

7.6 Applicability

This policy is applicable to all software applications/ components/ products whose IPR are held by any Government entity and which the concerned Government entity considers suitable for making the source code public. This policy will be in force for all software application development exercises initiated after the effective date of this policy. This policy will apply to all software application development efforts, whether in-house or through a software development agency. Applicability of this policy on software applications/ components developed prior to the effective date of this policy is desirable but not mandatory. Any procurement exercise for software application/ component/ application development services should give due consideration to this policy and the intent behind it.

This policy is not applicable on software applications/ components/ products utilized or implemented for projects/organizations of national strategic importance and for those projects / applications that may have security implications. The policy does not apply to Commercial off the Shelf (COTS) software.

7.7 Policy Statement

Government of India shall adopt uniform policy towards collaborative application development by opening the source code of Government applicationsto ensure a new and agile way of developing software, reuse and rapid roll out to other Government domains.

1. The Government will have full rights to custom-built software source code for any application developed by any Government agency or by private agencies funded by the Government.
2. If it is a COTS product, then the Government will have full rights on any customization code on the COTS product if it is procured by the Government. The Government shall have the right to reuse the customization code for any other Govt. department or entity if required. What components/code/modules constitute 'customizations on COTS' will be specified in the contract between theGovernment and the agency doing the customization.
3. If any agency customizes the source code or adds any modules or plug-ins to a particular Government custom-built application or customization code on COTS, the Government reserves the full rights of the source code of the add on modules, plug-ins or customization code.
4. In case an already successfully running application in the Government, whose code is opened and whose IP is owned by Government, needs to be rolled out by private agencies on a commercial basis for any other Government Department/Agency, code changes to the application source code is permissible, but Government reserves full rights to the source code of the modified application.
5. All Government application source code to be developed will be shared on the Collaborative Application Development Platform. To provide an effective and reliable platform for open source development, this Collaborative Application Development Platform shall have proper control mechanisms, version management and policies on verification/validation of the codes w.r.t required functionality, security, performance, design, coding practices and other necessary attributes.
6. While evaluating any new software for development or purchase by any Government entity, preference should be given to software already available in the Collaborative Application Development Platform.
7. The policy does not mandate already developed monolithic applications to

open their application source code on the Collaborative Application Development Platform, However, it is recommended that the application source code and the object code with installation script, installation document, database schema and any other documents be shared in the Collaborative Application Development Platform after due quality and security checks will be laid in the guidelines by Government of India. These guidelines would also address governance framework, operational processes, application maturity assessment models, application sustainability models, licensing policy etc. for the Collaborative Application Development platform.

8. The future procurement processes of the Government projects should ensure that the Government receives the **source code and unlimited rights** of custom-built application development. In case of COTS product, the contract clauses should secure full rights to customisation code developed on the **Commercial Off The Shelf** product. The rights should cover reuse of customization code anywhere else in the Government or public sector.
9. The policy does not restrict/prohibit any private/Government entity’s commercial interest either in development or implementation and support of Government applications. The commercial terms can be worked out between the concerned Government Departments and Agencies and solution providers on mutually agreed terms. The policy only that the application source code be opened for larger interest of rapid roll out and value addition to the application software through collaborative approach of development.
10. The policy does not impose any obligation on contributors to the source open Government projects to provide support if the application/component is downloaded for reuse by others. It is recommended that contributors should help others in improving the code or during its re-use, but this is not binding.

7.8 Responsibilities

#	Stakeholders	Role	Actions Required
1.	Department of Electronics & Information Technology Ministry of Communications and Information Technology	Facilitator	• Overall Guidance & Funding for the project • Issuing Policy on Collaborative Application Development by Opening the Source Code of Government Applications • Issue of Guidelines
2.	NIC (System Software Division)	Implementation Agency for Platform	• Awareness & Promotion • Application Owner Onboarding • Setting up, ownership and operational management of Collaborative Application Development Platform.
3.	Ministries/ Departments/ NIC/ CDAC etc.	Application Owners/ developers	• Compliance to "Policy on Collaborative Application Development by Opening the Source Code of Government Applications" • Release of source code of existing stable applications as per policy • Usage of Collaborative Application Development Platform for application development • Contribution to projects listed/ published on Collaborative Application Development Platform

4.	Ministries/ Departments	Application Seekers	• Adoption of applications published on Government collaborative Application development platform • Contributing to application modifications/enhancements • Compliance with the Policy
5.	Recognized Software Developers/ Academic institutions	Contributors	• Active involvement in platform usage and enhancement. • Testing of published application and publishing known vulnerabilities.

7.9 Review of the Policy

The Government shall have the right to revise the Policy as and when required.

7.10 Point of Contact

All queries or comments related to this Policy shall be directed to JS (e-Governance), DeitY (jsegov@deity.gov.in), Department of Electronics and Information Technology, Electronics Niketan, 6 CGO Complex, Lodhi Road, New Delhi – 110003 .

How will it be implemented?

- Roles of DeitY, Ministries, NIC, CDAC, recognized software developers / academic institutions have been defined in terms of their contribution in implementation
- DeitY shall facilitate implementation, NIC shall be the Implementing Agency for platform, Ministries/ Departments to be application owners and application seekers

Policy On Collaborative Application Development by Opening the Source Code of Government Applications

Most of the e-Governance solutions face long gestation periods in application development leading to unwanted delays in their implementation. This is a cause of concern as this leads to implementation of sub-optimal solution since the time by which the solutions are implemented, many technologies and processes undergo changes.

Collaborative application development by opening the source code of Government applications aims to transform the way application development is being done in the country. Since the source code of Government application would be made available, developers can improve such applications by best coding practices, leveraging latest technologies and improving user's interface. The policy aims to reduce the time for application development substantially so that e-Governance solutions could be implemented and rolled out on a fast track basis. Once the ecosystem of collaborative application development gets established and starts delivering results, the common citizens would be able to avail their services commensurate with the schemes announced by the Government in a time bound manner.

Application Development & Re-Engineering
Guidelines for Cloud Ready Applications

Application Development & Re-Engineering Guidelines for Cloud Ready Applications

This is meant to promote e-Governance solutions as cloud enabled products that can be utilized by various departments without having to invest time, cost and effort in application development. This would also help in rapid replication of successful e-Governance solutions.

The guidelines aim to ensure development of Common Application Software (CAS) which can be configured as per different state's / department's requirements without the need of modifying the core source code of the application for faster deployment.

The policy provides details on the following aspects:

- Software and Re-engineering guidelines
- Cloud enablement
- Self-Assessment checklist

Chapter 8: Application Development & Re-Engineering Guidelines for Cloud Ready Applications

8.1 Introduction

Productized and Cloud enabled applications are ideal solutions that can be utilized by various departments at centre and states without having to invest time, cost and effort in development of the same. This would enable re-use and deployment of applications rapidly across several states/departments.

8.1.1 Need for Software Development & Re-Engineering Guidelines

The basic need for Software Development and Re-engineering Guidelines is to ensure development of Common Application Software (CAS) which can be configured as per different states / departments requirements without the need of modifying the core code of the application for a faster deployment so that time, effort and costs in developing applications are saved and to obviate duplication of efforts. It is therefore imperative that applications are developed in conformity to guidelines that makes them standardized and compatible for hosting and running across states. This need has translated in the conceptualization, development and roll-out of productized cloud enabled application which can be centrally run & hosted and are available to states for configuring them as per their relevant processes with minimal customization for rolling out the services in shortest time possible.

It is envisioned that an application which is centrally run as a SaaS is easy to roll out to all interested parties at the same time and therefore such application’s architecture and design should be compliant to common minimum practices / considerations that will convert it to standard product.

What is this guideline?

It aims to facilitate cloud enabled products that can be utilized by various departments leading to savings in time, cost and efforts

8.1.2 Evolution of eGov App Store

The productized and cloud enabled application for states / departments will be made available on the eGovAppStore. The eGovAppStore was launched by the Hon'ble Minister of Communications & Information Technology, on 31st May 2013. The eGovAppStore is a national level common repository of customizable and configurable applications, components and web services that can be re-used by various government agencies/departments at Centre and States, with the vision to accelerate delivery of e-services as envisaged under NeGP and optimizing the ICT spending of the government with the following objectives:

- Speeding up the development and deployment of eGov applications
- Easy replication of successful applications across States
- Avoid duplication of effort and cost in development of similar applications
- Ensure availability of certified applications following common standards at one place

The key benefit for Stakeholders is that they need not reinvent the wheel and an application which is successfully running in another state can be made available to them expeditiously with requisite customization. Core and common applications that have high demand and are replicable across the central and state levels are the likely candidates for the eGovAppStore, which shall be hosted on the National Cloud. The eGovAppStore will include the setting up of a common platform to host and run applications (developed by government agencies or private players) at National Clouds under Meghraj, which are easily customizable and configurable for reuse by various government agencies or departments at the central and state levels without investing effort in the development of such applications

8.2 Software Development & Re-Engineering Guidelines

8.2.1 Solution Architecture

The solution architecture is key differentiator for product like solutions. A well architected solution gives it robustness for reusability (in code, configurations, databases, services etc.), enhancements and interoperability.

The following should be adopted as good architecture principles:

- **Well established Service Contracts**

  A contractual agreement between the Application Owner (Govt. Department at Centre/State or any Private Player) and the Application Provider (Govt. Department or independent entities which host & provide services through eGovAppStore) over the period of Application Lifecycle (for example: Productization + Replication + Hosting + Operation & Maintenance). The contracts related to licenses, source code etc. will also be a part of such agreements.

- **Loose Coupling of Services**

  This is one of the fundamental concepts of Service Oriented Computing. Loose coupling ensures that application components are treated individually and dependencies are reduced. This further ensures that addition, removal, failure or update of one component has a minimum impact on other components.

  Effort should be made to develop components separately and then their integration/ interaction mechanism could be defined in a separate component. For example, while developing a component that calculates the order of a commodity should not start calculating the total cost of the order placed. Order should be calculated separately and the cost should be calculated separately so that any change in costing structure should only affect the cost calculation code and not the order placement component.

- **Service Reusability**

  For the purpose of reusability, services should be written in such a way that they can be automated for testing. Test automation is necessary to ensure services can be upgraded, re-factored, etc. without breaking other services that use this.

  Further, all services should be inherently versioned and all invocations must specify the version of service. Efforts should be made to ensure that new versions of services should be backward compatible with at least one or two previous versions so that users of the service can start using new version of the service without mandatorily making changes to their code.

  Rapid Replication and productization of successful applications running across different States/UTs would ensure that these applications are also reusable in other states with appropriate built-in configurations which can be undertaken by concerned seeker state / department. The solution should also support minor customization if so essentially required by the seeker state / department. A repository of re-usable components is to be maintained and made available on eGovAppStore.

Software components can often be classified according to reusability levels:

- **<u>Foundation Components</u>**

  Examples of foundation components are classes such as Money, Date, List, Person and Number. These can be reused in almost any application
- **<u>Domain Components</u>**

  Examples of domain-specific components include classes like Customer, Account, and Transaction
- **<u>Architectural Components</u>**

  Examples of architecture-specific components include event notification mechanisms; user interfaces components, and message passing systems
- **<u>Application Components</u>**

  Examples of application-specific components include message handlers, exception handlers, and views.

➢ **<u>Service Abstraction</u>**

Abstraction provides control on what part of the service logic of a particular application are private (hidden) and what parts are made public (consumable). The public or consumable parts of the service logic can be designed in a generic manner to ensure that they encourage reusability as discussed in the point above. Abstraction also supports the loosely coupled principle discussed above. In a three tier (database, business and presentation) software application, necessary abstractions should be done in each layer so as to achieve loose coupling and to keep the code modular so that addition of any logic could easily be done at any tier. For example, in application development for scholarship disbursement system, a function to fetch beneficiary details may be designed to interact with database layer and gives the information to presentation layer. How the database layer performs the operation to fetch details should be abstracted from the business layer. Similarly how the presentation layer represents the information should be abstracted from the business layer.

- **Service Discoverability**

  While productizing the existing application or designing a new application for hosting on the eGovAppStore, it is important that accidental creation of redundant services or implementation of redundant logic is avoided. Service discoverability makes this happen by ensuring that metadata attached to a service and describes overall purpose of the service and its functionality, which makes the services easily discoverable. A repository of re-usable business logic components is to be maintained and made available on eGovAppStore. For example an existing service or business logic already available at the data center should not be recreated to save duplicity.

- **Service Autonomy**

  In addition to the principle of Reusability discussed above, it is important to ensure that services which are delivered do not just possess reusable logic, but they are also autonomous to be reused. This Autonomy will also facilitate adaptation to changing constraint in terms of scalability, service levels adherence, availability etc. For example only loosely coupled services or service components can be reused, therefore autonomy becomes an important parameter to efficiently design solutions.

- **Service Location Transparency**

  This refers to ability of the Service Consumers to use a service regardless of its actual location, for example being available on a cloud.

- **Service Granularity**

  Service Granularity means identification of optimal scope of business functionality in a service operation. Each service operation should ideally perform single transaction to simplify error detection, error recovery, and simplify the overall design (this means that particular Service operation is granular). In addition, each service operation maps to a single business function, although if a single operation can provide multiple functions without adding design complexity or increasing message sizes, this can genetically reduce implementation and usage costs (here each service operation is generalized enough and interoperable for multiple functions, making it granular).

- **Platform & Database Agnostic**

  From an architectural perspective, it would be required that the productized solutions should be not only be modular in nature, but be adaptive to converse with other technology components such as platforms and databases, complete with management suites or with the induction of adaptors and interfaces or even smaller bespoke solutions to support the same. It would also be required that the application provider should be able to deliver application on latest IT Infrastructure & system software components available at National Cloud and at SDCs under Meghraj. This would ensure that the applications developed can overcome the technology dependences and be available to a variety of seeker states.

- **Application design for occasionally connected systems**

  For the small percentage of functionality that requires “occasional disconnected/offline” operations, applications may be designed to use a local persistent store/cache just for the purposes of offline capability and later sync as and when connectivity is restored. As connectivity becomes ubiquitous, less of such offline capabilities are needed.

8.2.2 Standards Adoption & Solution Engineering

There are a number of standards available on software engineering lifecycles which ensure quality product development and scope of continuous improvements. The standards are to be followed as per the Government of India issued policies and guidelines promulgated from time to time.

The proposed solutions should be adaptable to the following as good software engineering practices:

- **Domain / Sector specific Meta Data Standards**

  Each sector or domain has its unique challenges in standardization of Meta-Data. It is important that any solution being developed to provide services in the domain or sector adhering to the Meta-Data standards for that particular sector or domain. This would ensure seamless integration between solutions developed for domain or sector. The GOI has also come out with Meta data standards which can be seen at www.egovstandards.gov.in

- **Software Engineering Standards**

  It is important that software engineering standards are adopted during the initial stages of the development lifecycle to ensure that the developed solution is able to meet quality certifications and security testing. Recommended testing requirements will be provided by STQC / empanelled agencies.

- **Usage of Open Standards technologies**

  As part of the software engineering, it is important to use technologies developed in open standards. As part of the overall software development lifecycle, a minimum customization and maximum configuration approach should be adopted. There should not be any hard-coding in any aspect of the development and release lifecycle of the proposed application. The following section articulates areas (no limited to) that should be available as configurable parameters, while overall software having the ability to be customized so as to meet the local requirements of the user state / department / agency. e-Governance application should preferably be developed using open source tools and components.

1. **Configurable Components**

An important facet of product like solution is its ability to be configurable to meet the business requirements. The following should be available as configurable components:

Master Data

Master data should be available in parameterized format. It should be based on the Meta data standards for the industry / domain / sector. They should not be hard-coded in the application.

- **Screen Labels**

  Screen labels may differ between solutions owing to the localization requirements for a solution proposed to be implemented. Configuration of screen labels should be made available through resource files. They should not be hard-coded in the application.

- **User Alerts & Messages**

  Based on the user departments business requirements, alerts and messaging services need to be pushed or pulled to the end user. Allowing for alerts and messages to be available as a configurable component would ensure that unwanted alerts and messages are not routed through to all workflow entities.

- **Reports**

  It is generally required from solutions to be able to prepare various kinds of reports for various levels of officers in the hierarchy, along with aggregation and data sorting features. Available as a configurable component, it would ensure that the reports are localized to the needs of a user, rather than being generic to business function or sub-unit.

- **Workflow Management**

  Common business functions in two similar organizations may have different processes related to approvals, escalations, reviews, recommendations etc.; therefore it is important that workflows are available as configurable components to allow the solution to be configured to the business requirements of that organization.

- **Multi Language Support**

  Government departments operate in multiple languages depending on their region. Product like solutions should be adaptable, to allow through configuration, selection of language in which the user wishes to operate the system. Product like solutions should at least be bi-lingual, with English as one of the languages.

- **Business Rules (if - then - else)**

  Business rules are at the core of workflow processes and allow for information, interaction and transaction services to be communicated. Product like solutions should ensure that business rules are configurable to allow the organization to localize the solution to their business requirements. They should not be hard-coded in the application.

- **Dashboards**

  As a management tool, most senior officers require dashboards to review service progress, service levels, escalations, alerts and reminders, messages etc. As an operational tool it is required by the office staff for work-list detailing, alerts, reminders and messaging. As configurable component, it would ensure that the user is able to see his or her, role based dashboard for summary of tasks and activities to be completed.

- **<u>Online Help & Feedback</u>**

  As a feature in most standard products it would be required that online help and feedback mechanism should be available as configurable parameters to assist the users in functioning of the application. This could include context sensitive help, user manuals etc. In online feedback mechanism, feedback on technical aspects as well as service delivery should be given to the users.

2. Customizable Components

A solution may be required to be customized to meet specific business requirements of an organization. The following should be kept in perspective while customizing core solutions:

- **<u>Ability to add additional features without compromising the core code</u>**

  The solutions should be developed in modular format, or should allow for modular integration or interfacing with other solutions, without the need of editing existing core code. Solutions should allow for the development of new features, functionalities, changes to done through interfaces external to the existing code base.

- **<u>Ability to interface with other independent sub-applications</u>**

  It may be required that a product like solution is required to interface with other bespoke smaller applications, unique to an organization. There should be minimal effort required for such activities, and should be made available through external adaptors interfacing with the core application.

  Methods of customization could include:

1. Implementing a plug-in architecture so that tenants could upload their own code through defined interfaces without changing the core application or;
2. using some form of rules engine that enables process customization through configuration
3. Another alternative to consider is enabling application to call a service endpoint provided by the tenant, which performs some custom logic and returns a result.

In addition, application may also require providing ways to extend the application without using custom code. To achieve this application must implement a mechanism for customizing the UI, and a way of extending the data storage schema.

Methods of extending schema can be:

- Single fixed schema with a set of columns available for custom data
- Single fixed schema with separate tables holding custom data

3. **Mobile Enablement**

The reach of mobile technology and devices has percolated beyond the last mile of connectivity into the households of the most unreachable terrain in India. Therefore it becomes important that government service delivery is undertaken through this medium to increase the scale and reach of government services throughout the nation.

As a resultant it is required that the applications that planned to deliver these services use the mobile medium to provide services. There are three means through which applications can be engineered to provide services through mobile enablement:

1. Accessing application on a mobile device
2. Accessing a mobile version of the application through a mobile device (m.website)
3. Accessing a mobile application through a mobile device

In the first means the accessibility of the application through the medium changes translating from a system based access to a mobile device. The second means assumes the redevelopment or reconfiguration of the application to suite a mobile based delivery platform, (m.website) which follows best practices in providing applications with limited or complete functionality to be accessible over a variety of mobile service delivery resolutions (such as in case of smartphones, tablets, key interface phones). The third means assumes the redevelopment of an additional application or app, which can be downloaded and run on the mobile device.

Furthermore the application access can be given through multiple means over mobile devices in formats such as USSD, SMS, App etc. It is predominantly decided the services being offered by the parent domain department / agency to select the means through which the services can be provided.

In case of native mobile application development wherein business layer is planned for deployment on remote tier, a separate service layer can be designed. Services should be designed for maximum reusability by not assuming any specific details of client. For improving interoperability, REST based protocols and transport mechanisms can be implemented.

From an application development and re-engineering guidelines perspective it is required that the applications are developed to meet mobile device service delivery platform requirements while at the same time ensuring security of data, ease of use of the application and continuation of the citizen experience as over traditional access mechanism.

8.2.3 Integration & Interoperability

A key requirement for any product is its ability to interface, integrate and more importantly be interoperable with other technology suites. The application should be developed in a manner that it should support flexible, modular and extendable services. The proposed solution should have the following:

- Clear input and outputs should be defined
- Ability to perform business validations
- Clearly defined error codes
- Support (i) Asynchronous (ii) Synchronous (ii) Batch mode, models of integration
- Support Web Services
- Support File Transfer
- Support SMTP
- Support Mobile (SMS) service delivery
- Support API based integrations
- Support Push & Pull Integration
- Support Published / Subscribed methods such as Java Messaging Service, RSS etc.
- Support integration on open standards
- All major validations / constraints such as primary & foreign keys can be at the database level and others such as business logic at the context level
- Access should be compatible with external devices such as hand-held devices, tablets & smartphones

1. **List of Open APIs proposed to be published**

The e-Governance projects are linked to each other because they service a common list of beneficiaries, i.e. the citizen. It is required that new applications developed and those reengineered should capture and process data limited to their agency / department. Any data which can be furnished or exchanged through an external department / agency should be done through the use of APIs (Application Programming Interface). This will allow the application to source data through a unified data pool and will marginalize errors in data entry for the same record. API invocation must allow platform neutral and language neutral way of calling. For example, a service written in Java should also be usable within an application developed in .NET environment.

The proposed application should list all the APIs that it intends to provide to be consumed by other departmental applications (including citizen interfaces) and should also list data elements which it needs to be sourced from other departments. Prime examples of this can be UID for personal information, Vahan data for vehicle data etc.

8.2.4 Quality Certification, Release Management & Documentation

1. Quality Certification

It is important for product like solutions to adhere to quality certification processes to ensure that solutions being given for replications to other stakeholders, meets minimum quality benchmarks. To ensure a quality product it would be required that the solution:

- ➢ Should qualify defined functional testing through STQC / empanelled agencies
- ➢ Should qualify defined performance testing through STQC / empanelled agencies
- ➢ Should qualify defined security testing through empanelled agencies
- ➢ Should have well documented development & testing process artifacts
  - o Business Requirements Document (BRD)
  - o Functional Requirement Specifications (FRS)
  - o Software Requirement Specifications (SRS)
  - o Software Design Documents (including HLD, LLD etc.)
  - o Requirements Traceability Matrices (RTM)
  - o Test Plan, Test Cases & Test Reports
  - o Code Review Reports
  - o Database Review Reports
  - o Project Implementation Plan User Manual
  - o Deployment Guide

Detailed testing requirements will be provided by STQC.

8.2.5 Solution Sizing & Scalability

Since the solution will be required to be hosted on various deployment models, it is important for the solutions to be able to scale up to meet increasing usage requirements. Although an initial estimation of the hardware specifications (quantity and model / version) would be required to size the solution based on system interaction, to increase capacities the solution should adaptable to scaling. The following should be kept in perspective:

- **Able to scale up to meet increasing load**

  Solution should be able to handle increasing number of first time users, transactions, data sharing processes etc.

- **Able to demonstrate stress levels exerted**

  Solution should be able to handle increasing number of concurrent users, concurrent transactions, synchronous data sharing with other systems etc.

- **Able to perform on throttled bandwidth environments**

  Solution should be able to perform to the agreed service levels regardless of the bandwidth available or in multiple bandwidth availability scenarios

- **Should have low technical & infrastructure resource consumption**

  Solution should optimally use technical resources such as memory, processor (CPU), storage etc. In addition should optimally use data center resources on available bandwidths.

- **Should be interoperable to newer technology upgrades**

  The solution should be able to harness the advantages of legacy technology (servers, software, devices etc.) while be able to upgrade to newer systems. This would enable low cost – optimal utilization of resources.

- **Horizontal Scalability**

  Scalability of an application is aided through designing services as granular as well as loosely coupled. Use of distributed data stores and sharding also aid application scaling. If the service uses database/datastore, it must ensure database layer can also span multiple database nodes

a. This can be achieved either by using a distributed data store; or

b. If using traditional RDBMS systems, this can be achieved by ensuring application level sharding (partitioning) is implemented to partition data across many RDBMS nodes. Each shard has the same schema, but holds its own distinct subset of the data. A shard is a data store in its own right, running on a server acting as a storage node.

8.2.6 Language & Interface

A key requirement for government application being available nationally is their ability to provide the user a local interface and support local language. Therefore the proposed solutions should:

Why we need it?

- To ensure development of Common Application Software (CAS)
- To save time, cost and efforts
- To obviate duplication of efforts

➢ **Be developed on Unicode Compliant Code practices**

The development should be undertaken using Unicode compliant practices.

➢ **Support open standards on language interfaces**

The solution should support open standards on language interfaces.

➢ **Should support multiple language (Indian & Foreign) APIs**

Solution should at least be bi-lingual, but should possess capabilities to be multi-lingual.

➢ **Should support self-learning data dictionaries**

The solution should be support APIs that enable building of transliterated data dictionaries, with preemptive text, so that the user is given the choice to select the nearest match.

8.2.7 Legacy Integration – Digitization & Migration

The proposed solution should be able to acquire, sort and store the data that has been accumulated for the service being provisioned through multiple legacy ICT solutions. Therefore the proposed solutions should be:

- **Able to migrate data through offline user interfaces**

  The solution should provide for manual data entry of legacy data (allow for conduct of digitization activities)

- **Able to migrate data through be-spoke / product utilities**

  Solution should support migration legacy data through be-spoke utilities which allow for data entry, extraction and submission of data into the proposed solution

8.2.8 Intellectual Property Rights (for Center & State owned applications)

- The Intellectual Property Rights for the developed product should invariably reside with the Government Department. This should include the source code, release management artifacts and all other technical and domain related documentation for the developed solution. The licenses procured for the implementation of the existing application may be provided.

○ Release Management Artifacts should include, but not be limited to the following:

- Core Application
- Packaged Installation
- Application Code
- Code Review
- Unit Test Results (Multilingual)
- Test Suites
- UAT Scripts & Test Cases (Multilingual)
- User Interface Testing Results (Multilingual)
- Performance Test Results
- Security Test Results
- Requirement Traceability Matrix
- Deployment Scripts
- Deployment Manual
- User Manual
- Technical Manuals
- Release Notes
- Standard Operating Procedures
- Application Customization Guidelines
- Quality Assessment Report
- UAT Acceptance Benchmarks
- Mapping sheet for defects/functionality and system test cases
- Non-Functional Requirements Compliance sheet
- Backup of the Database before executing the incremental Script
- Incremental Script
- Release note for Database changes done between builds
- DB Code Review Report

- The IPR for the developed product / solution should not be restricted / compromised through any legal interpretation. The solution should clearly be the property of the government department.

8.3 Cloud Enablement of Applications

8.3.1 Application Migration to Cloud

There are five well established approaches to migrate traditional applications to the cloud, these include:

1. **REHOST** on Infrastructure as a Service (IaaS)
2. **REFACTOR** for Platform as a Service (PaaS)
3. **REVISE** for IaaS or PaaS
4. **REBUILD** on PaaS
5. **REPLACE** with Software as a Service (SaaS)

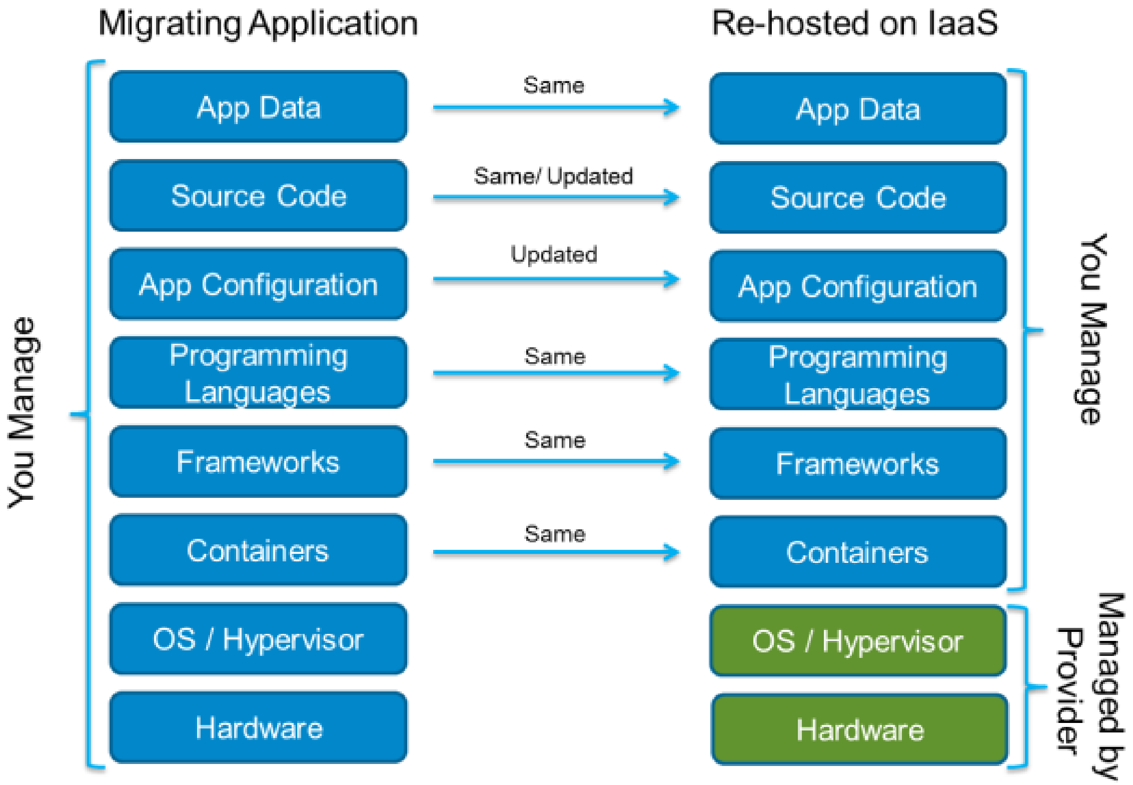
How will it be implemented?

- Adoption of architecture principles like service contracts, loose coupling of services, service reusability, service abstraction, service discoverability, service autonomy, service granularity, platform and database agnosticity
- Implementation of software engineering practices like metadata and data standards and open standards technologies
- Provision of configurable components and customizable components
- Mobile and Cloud enablement
- Integration and interoperability through open APIs
- Quality certification, release management and documentation
- Legacy integration – digitization and migration
- Cloud enablement through approaches like **REHOST** on Infrastructure as a Service (IaaS), **REFACTOR** for Platform as a Service (PaaS), **REVISE** for IaaS or PaaS, **REBUILD** on PaaS and **REPLACE** with Software as a Service (SaaS)
- Application Self-Assessment checklist

1. Rehost on IaaS

This approach involves the re-hosting of the application from the existing infrastructure to the cloud infrastructure without making any significant changes to application code-base or the application configuration files. The application Operating System / Hypervisor and the Hardware in addition is managed by the cloud provider.

The following diagram depicts the re-hosting of the application on Infrastructure as a Service.



Rehosting solutions vary from a hosting infrastructure to application virtualization.

Example: There are ways by which server applications are moved rapidly to and across the cloud, without code change or lock-in. Use of toolkits may be made that allow cloud integrators to handle migrations on behalf of their enterprise accounts.

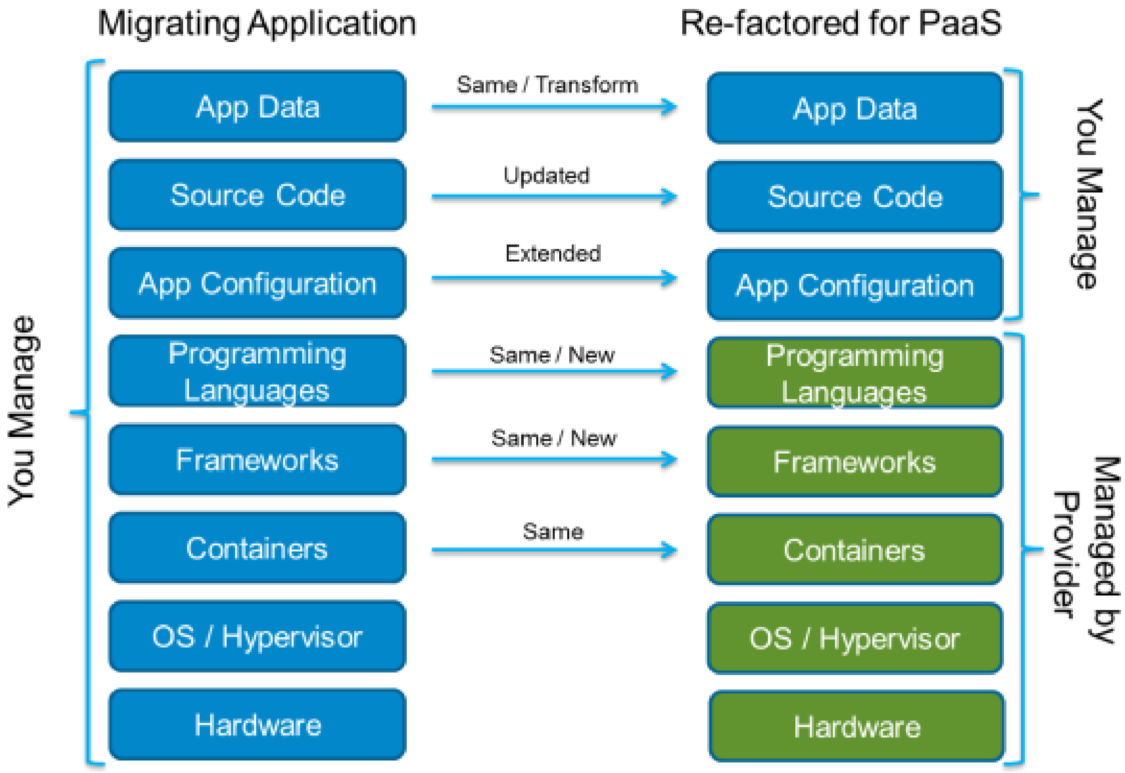
Taking an application-centric approach in moving Server applications, use of application images rather than server or machine images is considered more efficient i.e. encapsulating an application and its dependencies in what is called a "virtual application appliance" (VAA), without a virtual machine (VM). The result is application flexibility that is hypervisor-agnostic, cloud independent, and fast.

(e) Refactor for PaaS

This approach involves the refactoring of the application to use the platform provided by the cloud provider to migrate the application. In this method the programming languages, development frameworks, containers, operating system / hypervisor and the hardware are all managed by the cloud provider.

In addition application data is kept the same or transformed upon migration, application source code is updated, application configurations are extended to service the customer and programming languages and development frameworks are either kept as the same or new ones provided by the platform are used.

The following diagram depicts the refactoring of the application for Platform as a Service.



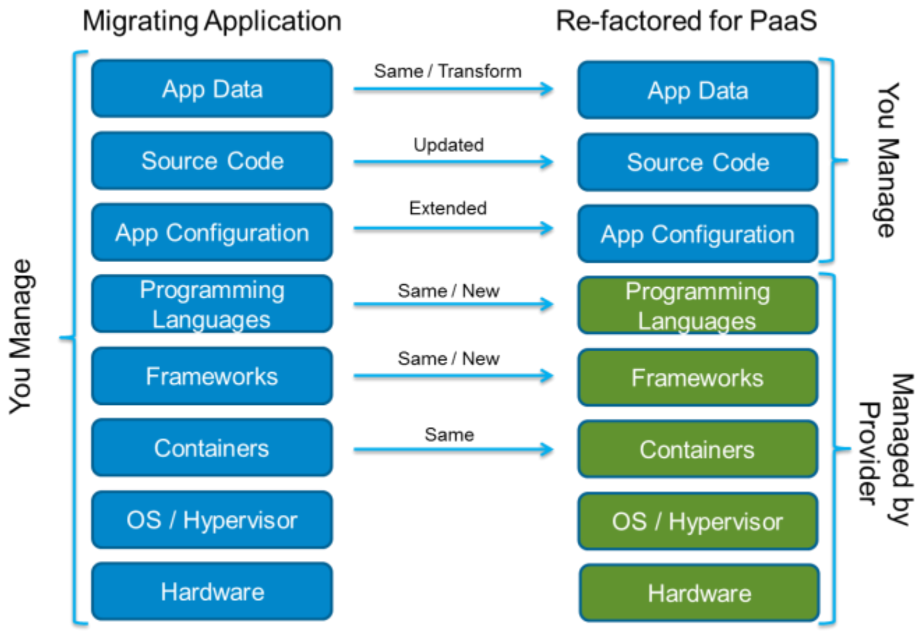
In simple terms, refactoring means doing just enough to migrate an application to a platform-as-a-service cloud offering. It is not merely lifting an application onto a PaaS, because the way the vendors handle security, authentication and data access is generally very different which leads to break open the code in order to use the new frameworks and libraries in the platform." So application code needs to be refactored to leverage the benefits of the PaaS frameworks.

(f) Revise for IaaS or PaaS

This approach involves the migration of the application requiring rebuilding the application utilizing either the infrastructure components of the cloud or utilizing the platform components of the cloud. In this approach similar to the Refactor approach, programming languages, development frameworks, containers, operating system / hypervisor and the hardware are managed by the cloud provider; while the application data, source code and application configuration is managed by the development agency.

In addition the application data is kept as a same or transformed, the source code is updated, new application configurations are required, same or new programming languages, development frameworks and containers are used for revising applications.

The following diagram depicts the revising of the application for Platform as a Service or Infrastructure as a Service.



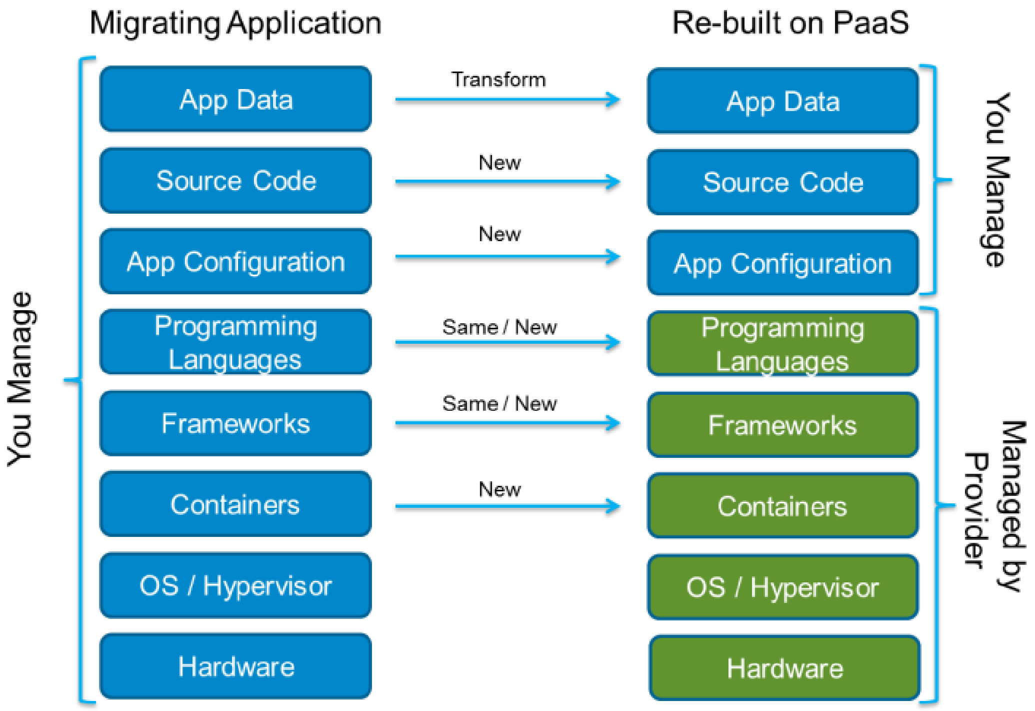
Refactoring does a minimalistic change in the application which is required to move it onto a PaaS system. But in order to reap the maximum benefits from the scalability of cloud infrastructure, the application has to undergo more fundamental changes to the architecture. The development team has to do very significant work in the application to make it cloud optimized. The guiding principle on whether this kind of revision is worth paying for should be the value of the code in question.

(g) Rebuild on PaaS

This approach involves the redevelopment of the application to suite cloud based deployment. Similar to the revise approach, in this approach also the programming languages development frameworks, containers, operating system / hypervisor and the hardware are managed by the cloud provider; while the application data, source code and application configuration is managed by the development agency.

In addition the application data is transformed from the existing infrastructure to the new environment, the source code and application configurations are written / configured anew. The existing or new programming languages and development frameworks from the cloud platform are used.

The following diagram depicts the rebuilding of the application on Platform as a Service.

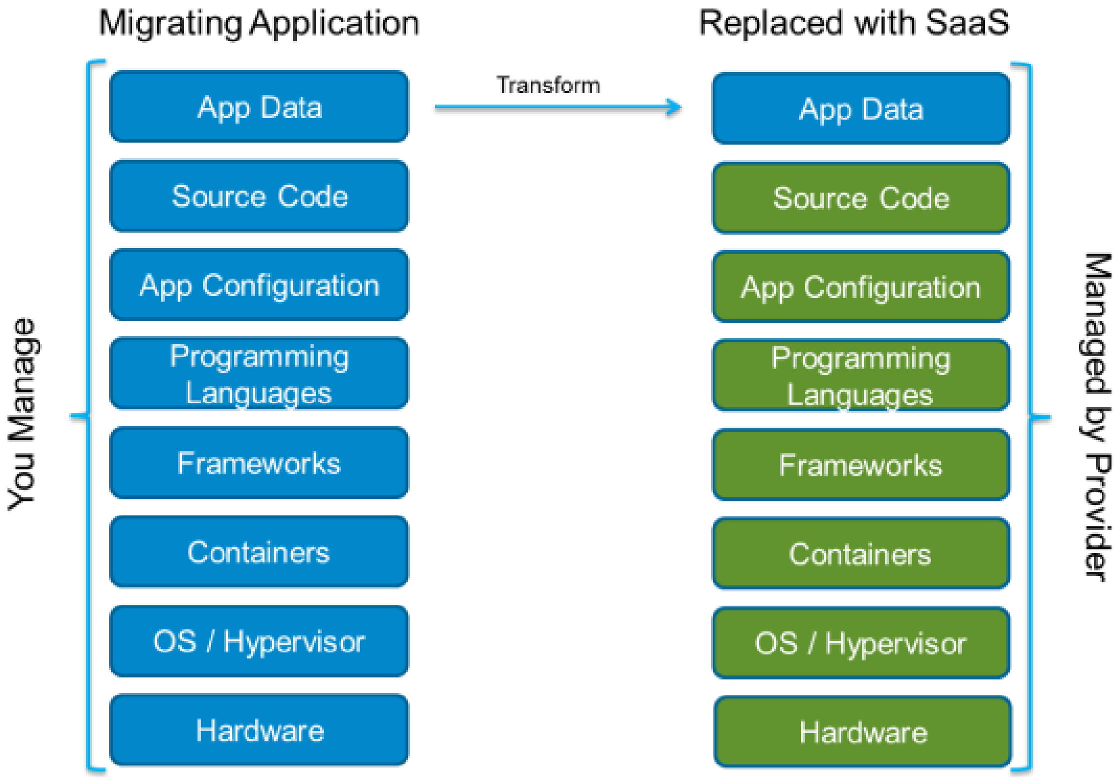


PaaS offerings include facilities for application design, application development, testing, and deployment as well as services such as team collaboration, web service integration, and marshalling, database integration, security, scalability, storage, persistence, state management, application versioning, application instrumentation, and developer community facilitation.

(h) Replace with SaaS

The last approach for migration of the applications to cloud involves replacing the existing application with a new application, which is completely managed by the cloud provider, and is available to the customer on Software as a Service Model. In this approach only the application data from the existing application is transformed to the new application; while all other aspects such as source code, application configuration, programming languages, development frameworks, containers, operating system / hypervisors and hardware are managed by the cloud provider.

The following diagram depicts the replacing of the application with Software as a Service.



In contrast with the other approaches, the replace approach suggests to use the SaaS solutions instead of building the application.

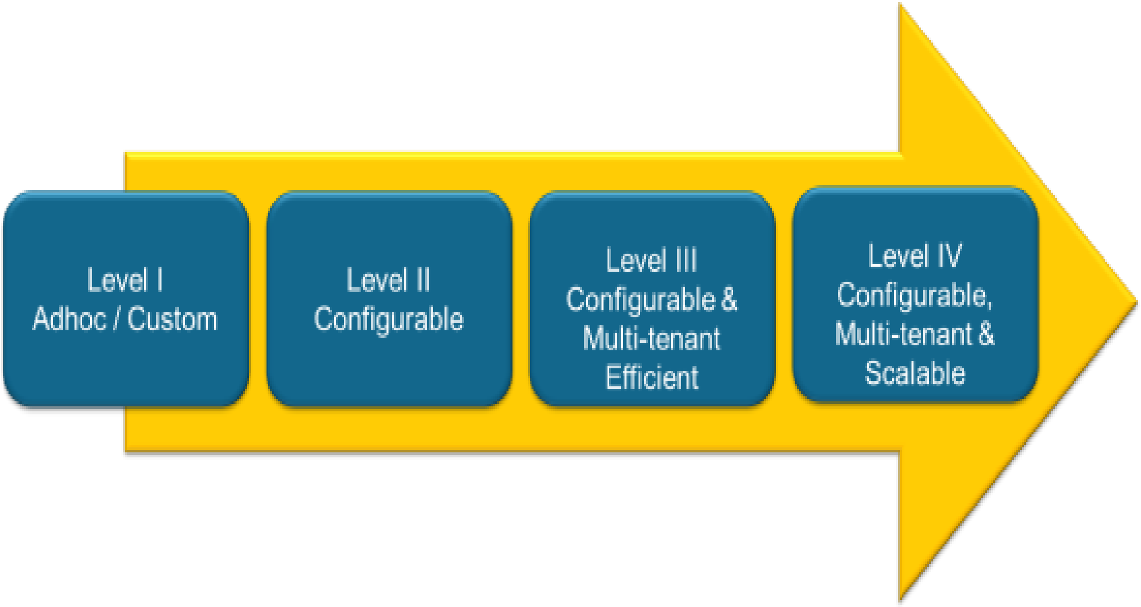
8.3.2 Software as a Service Characteristics

The following should be considered as key aspects for development of applications planned for deployment on SaaS model:

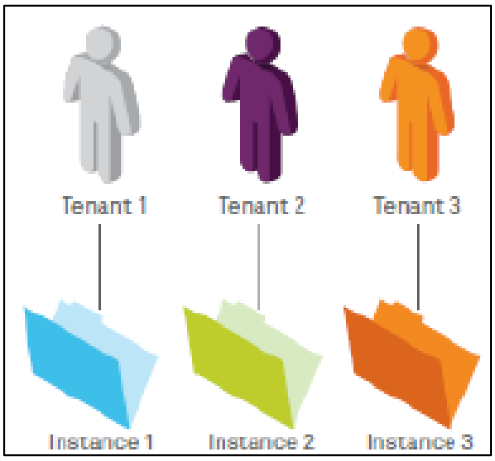
- The application should support **Multi-Tenancy**
- The application should have certain level of **Self-Service Sign-Up**
- The application should be **Scalable** in nature
- The application should be **Stateless** in nature
- The application should support mechanisms to **Measure Service**
- There should be a mechanism in place to support **Unique User Identification & Authentication**
- There should be a mechanism in place to support **Configurability** (UI, Business Logic, Workflow etc.) for each tenant
- There should be functions in place to **Monitor, Configure, & Manage** application & tenants

1. **SaaS Maturity**

The following diagram depicts the various stages of SaaS maturity model:

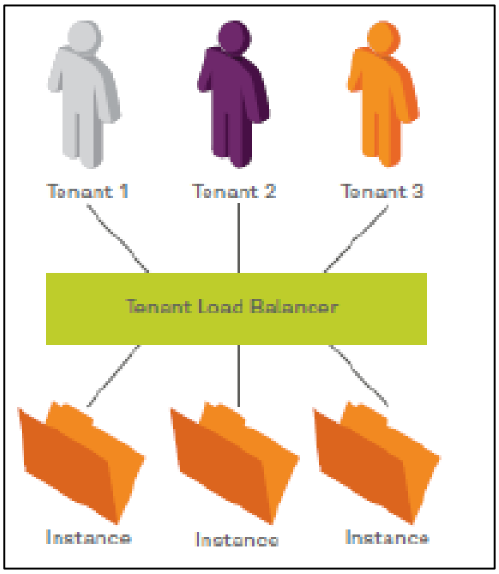

Level I
Adhoc / Custom
Level II
Configurable
Level III
Configurable & Multi-tenant Efficient
Level IV
Configurable, Multi-tenant & Scalable

SaaS Maturity Level I



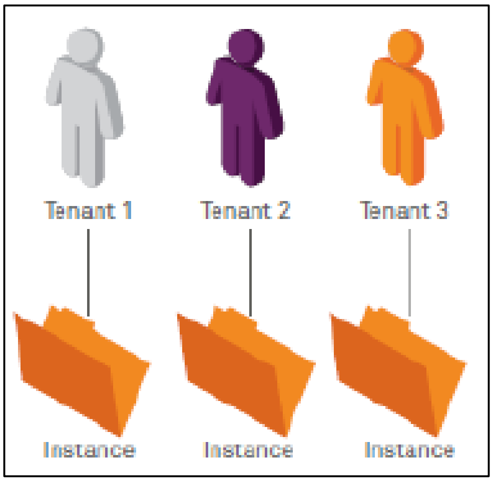
Level I of the SaaS maturity model implies that custom development is undertaken for each tenant and managed separately. In such instances the application development agency has to manage the changes being suggested for each tenant along with version control for each deployed version of the application.

SaaS Maturity Level II



Level II of the SaaS maturity model implies that multiple copies of the same instance are run separately each tenant. In such a model the application development agency has to manage configuration files for each tenant separately and make changes based on the business requirements.

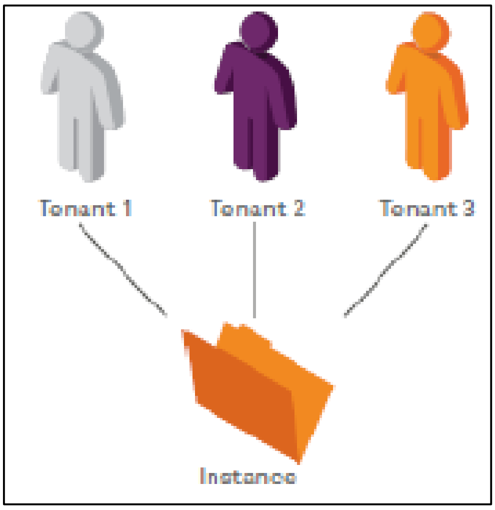
SaaS Maturity Level III



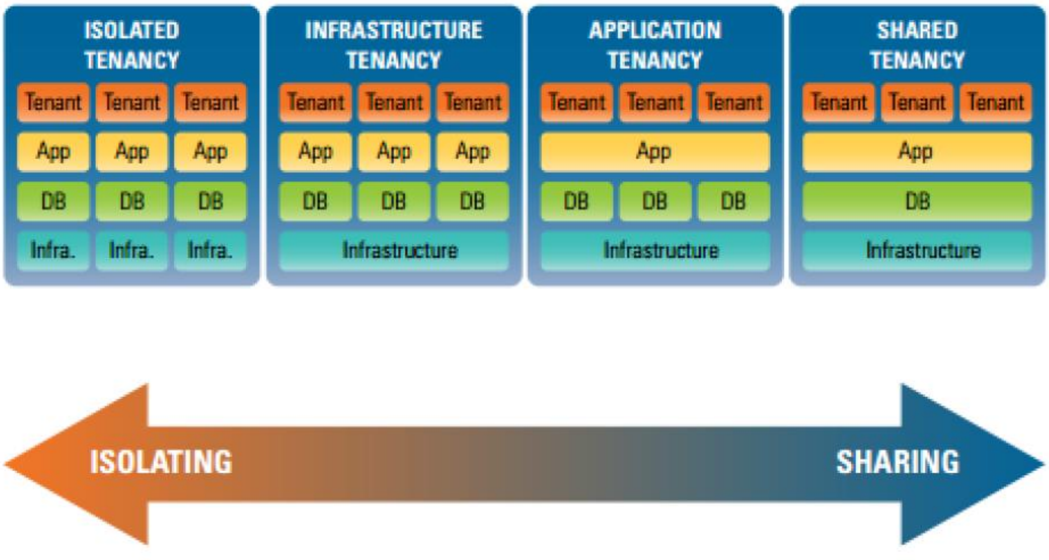
Level III of the SaaS maturity model implies that the single instance of the application is used by all the tenants. Different configuration files are managed through a multi-tenant efficient architecture. The configurations are managed by the tenant themselves and can be changed in run-time.

SaaS Maturity Level IV

Level IV of the SaaS maturity model implies that multiple instances of the application can be managed through a tenant load balancer, which creates additional instances based on the load on the application. These instances serve a number of tenants and are based on the configuration files defined for the application.



2. Multi-tenancy



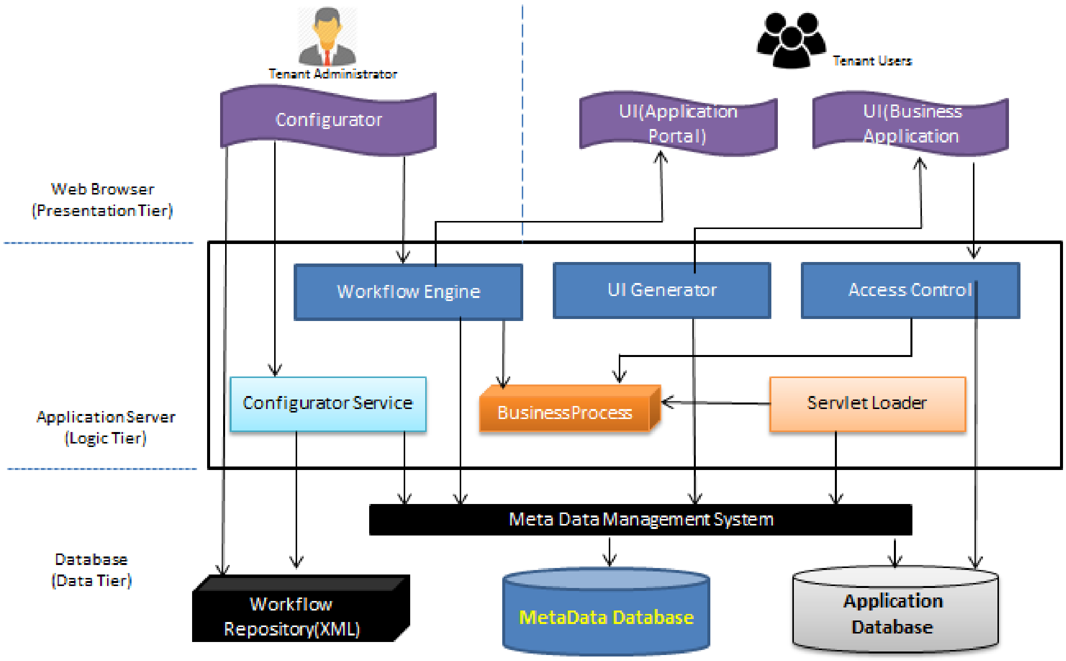
Multi-tenancy is defined as an architecture in which a single instance of an application serves multiple customers. The following diagram depicts the multi-tenancy continuum from isolated tenancy to shared tenancy utilizing the cloud resources.

In order to develop multi-tenant application the architecture should be designed and developed in a manner so as to have the following:

- Identification of most granular functionality
- Implementation of functionality as a web service
- Orchestrating each functionality to configure the desired workflows
- Configuration of application workflow using workflow designer
- Execution of application workflow using workflow engines
- All configurable aspects of application should be stored in separate tenant specific metadata database
- UI (User Interface) Customizability
- Segregated data storage (tenant-wise) for protecting access and data isolation amongst various tenants

3. Designing Configurable Application

The following diagram depicts architectural maturity for all application development envisaged to be made available under the national eGovAppStore. The architecture proposes applications to be built as component stores, which will allow the application to be readily re-usable and scalable, two of the key design aspects for cloud enablement.



Application Tiers

The application should be developed on a multi-tiered architecture to benefit for distributed computing and scaling advantages brought onboard by cloud enablement. These should be primarily split into Presentation, Application and Database Layers. These distinct layers can subsequently be divided into multiple sub-tiers which will allow for greater ease of simultaneous computing capabilities to be made available on persistent infrastructure.

Application Design

The application should be designed in a manner that provides clear distinction between configuration components and execution components. This will allow the application to scale-up in run-time to handle more simultaneous service requests, while at the same time allow the administrators to manage configurations for multiple tenants. Both configuration and execution processes should be developed as stores loosely coupled to allow better access to processes within. Process stores of architecture design should have User Interface Configurator, Service Configurator, Workflow Configurator, Business Logic Configurator, Meta-data Management System, Access Controllers, Run-time Engine & Application Database.

8.4 Annexure I - Application Self-Assessment Checklist

The following check-list needs to be self-assessed by Application Development / Providing / Provisioning Agencies to be eligible for the proposed national level App Store.

Application Details	
Application Name	
Application Current Version	
Released on	
Certification (if any, done by)	
Provider department details	
Initial cost of application development	
Proposed effort and cost of application customization for Cloud enablement (in case the application is not cloud enabled)	Effort: Man Months Cost: INR

Rating Criteria

Rating 1 – The application is non-compliant and cannot be changed

Rating 2 – The application is presently non-compliant and would require more than 50% of the original development effort to change the application

Rating 3 – The application is presently non-compliant and would require between 30% to 49% of the original development effort to change the application

Rating 4 – The application is presently non-compliant and would require between 10% to 29% of the original development effort to change the application

Rating 5 – The application is presently non-compliant and would require less than 10% of the original development effort to change the application

Rating 6 – The application is fully compliant on the component

S No	Compliance Component	Rating (between 1 – 6)	Weight	Provide Details
1.	Is master data, screen labels, user alerts, reports, dashboards configurable (not hardcoded in the application)		4	
2.	Are business rules configurable (managed using rule-set engines and not hardcoded in application)		4	

3.	Are workflows configurable (not hardcoded in the application)		4	
4.	Is user interface (application screens - look & feel) customizable		4	
5.	Can the application be integrated with SMS Gateway and/or Messaging Systems		5	
6.	Can the application be integrated with other external applications / components / services, implies: (i) Application can be integrated with payment gateway, third party applications etc. (ii) Application can be integrated with third party components such as Identity & Access Management Tools etc.		4	
7.	Can the application be integrated with other external applications / components / services, implies: The application has defined interface points and mechanism for data exchange		4	
8.	In case required, is the application developed in such a manner that it can support offline data entry & synchronization mechanisms.		2	
9.	Is the application designed on a 'Multi-Tiered' architecture, implies: (i) Are tiers configurable with minimal effect to other tiers (ii) Is there clear segregation of duties between presentation, business and database layers		4	

10.	Is the application scalable		5	
11.	Is the application deployable on multiple platforms		1	
12.	Is the application developed on Service Oriented Architecture		3	
13.	Is the application deployable on multiple databases		3	
14.	Can the application be deployed as packaged installation and creates verification log for the installation		2	
15.	Does the application have multilingual capabilities, implies: that the application is UNICODE compliant		3	
16.	Can new features be added in the application from a remote central location, implies: application supports automated patch management		3	
17.	Is the application developed in a manner that in case newer versions of the core application are released, it does not affect the integrated components to the core application.		4	
18.	Are the application release management, configuration management and version management clearly articulated with well-defined policies, implies: project artifacts such as SRS, FRS, RTM etc. are available.		5	
19.	Is the application developed in a manner that it is multiple browser compatible including backward compatibility of bowsers		3	

20.	If required, is the application accessible through multiple clients including handheld devices, tablets, smartphones etc.		3	
21.	Does the application support Multi-Tenancy?		5	
22.	Is the application designed to store configuration files outside the application & allowed to be changed in run-time?		4	
23.	Is the application designed to be Scale-Out or Scale-In?		5	
24.	Is the application designed to be Stateless?		5	
25.	Does the application assume any specific infrastructure dependency?		3	

8.5 Definitions

S No.	Keyword	Definition
1	**Product**	A well-developed product is defined as an integrated packaged solution which is available to the end user for ready to use. This proposed solution may require configurations to adapt to the business processes of the end user. Also the product should only allow for minor customizations to localize the solution for end user department / agency.
2	**Service Contract**	A service contract is defined as a physical contract which is signed between the service provider and service consumer. In context of the project service provider would imply, the App providing department / agency / stakeholder and the service consumer would imply the department / agency that would be using the product.
3	**Loose Coupling of Services**	It is explained as concept wherein the individual services designed, developed and integrated as part of the solution are loosely coupled in the solution, so that in case another solution, service wants to use / re-use the service they are able to do so.
4	**Service Reusability**	It is explained as a concept wherein the individual services designed under a solution for a particular business unit can be reused by configuring certain parameters to suite the business requirements of another business / functional unit.
5	**Service Abstraction**	It is explained as a concept wherein the application development takes account of the reusability factor and allows for the developed service and its components to available for other services / solutions. This would also ensure that there is transparency in the development of the application and that features can be reused as part of other services.
6	**Service Discoverability**	It is explained as a concept wherein services loosely coupled under a solution are easily identifiable, so that other services / solutions do not replicate the logic defined in another service. This would help in weeding out redundant logic in the developed application code and make its performance optimized.
7	**Service Autonomy**	It is explained as a concept wherein the services which are loosely coupled, discoverable and can be easily abstracted are single as far as their development is concerned. It implies that the functional logic that has been developed for the service is not required to be replicated in as part of another development in the

S No.	Keyword	Definition
		same solution.
8	**Service Location Transparency**	It is explained as a concept wherein a service that has been developed under a solution, is not only discoverable to other solutions / services, but is also denotes clearly its location, such as locally available, available on the networked data center or available on the cloud. This would ensure that the end user / solution are able to use the service independent of its physical location.
9	**Service Granularity**	It is explained as concept wherein each service is developed in a manner that it easily understood, used by other services, and that the actions performed under the services are transparent. This would include the maintaining the activity logs – including user information, delta change, time stamps,
10	**Platform & Database Agnostic**	It is explained as concept wherein the solution is developed in a manner that it has ability to integrate with other systems, developed diverse platforms, in addition to being interoperable with multiple databases available. This would ensure that service delivery is the prime forte of the solution, while technology and infrastructure support its delivery.

The maturity level of e-Governance in the country varies from States to State. Some States are advanced in implementation of their e-Governance projects while other states are facing difficulties and are still to make the optimum use of IT in their processes and citizen centric service delivery. As an outcome of the revamping exercise of mission mode projects / e-Governance initiatives like Road Transport, PDS, e-Courts, e-Prison, Treasury, CCTNS etc, it has been notified that there are many cases, wherein, multiple versions of the same solution are running in various States, each with suboptimal performance. This is a genuine concern as it causes cost overrun, unwanted delays and duplication in efforts. These guidelines aim to address the aforesaid issue through development of Common Application Software (CAS) which can be configured as per different state's / department's requirements without the need for modifying the core source code of the application. This would facilitate faster deployment and would also save time, efforts and costs.

References

References

S.N.	List of Gazette Notifications and Office Memorandum
1	**Chapter 1: The e-Kranti Framework**
	Status: Notified through Office Memorandum 5(12)/2015-EG-I dated: 8th May, 2015 Subject: Approval of Approach and Key Components of e-Kranti: National e-Governance Plan (NeGP) 2.0 http://deity.gov.in/sites/upload_files/dit/files/Office%20Memorandum%20on%20e-Kranti.pdf
2	**Chapter 2: Policy on Adoption of Open Source Software for Government of India**
	Status: Notified through Gazette Notification vide REGD. No. D.L.-33004/99 No. 79] dated: 27th March, 2015 Subject: “Policy on Adoption of Open Source Software for Government of India”
3	**Chapter 3: Framework for Adoption of Open Source Software in e-Governance Systems**
	Status: Notified through Gazette Notification vide REGISTERED No. DL(N)-04/0007/2003-05 No. 20] dated: 16 May – 22 May, 2015 Subject: “Framework for Adoption of Open Source Software in e-Governance Systems” https://egovstandards.gov.in/system/files/PublicReviewDocument/Framework_on_OSS_Ver0.8.pdf
4	**Chapter 4: Policy on Open Application Programming Interfaces (APIs) for Government of India**
	Status: Approved and is in process of Gazette Notification Subject: ““Policy on Open Application Programming Interfaces (APIs) for Government of India” http://deity.gov.in/sites/upload_files/dit/files/Open_APIs_19May2015.pdf
5	**Chapter 5: Email Policy of Government of India**
	Status: Notified through Extra Ordinary Gazette vide REGD. No. D. L.-33004/99 No.

	44] dated 19.02.2015 Subject: “E-mail Policy of Government of India” http://deity.gov.in/sites/upload_files/dit/files/Gazette_notification_of%20E-mail_Policy_of_Government_of_India.pdf
6	**Chapter 6: Policy on Use of IT Resources of Government of India**
	Status: Notified through Extra Ordinary Gazette vide REGD. No. D. L.-33004/99 No. 45] dated 18.02.2015 Subject: “Policy on Use of IT Resources of Government of India” http://deity.gov.in/sites/upload_files/dit/files/Policy%20on%20use%20of%20IT%20resources%20of%20Government%20of%20India.pdf
7	**Chapter 7: Policy On Collaborative Application Development by Opening the Source Code of Government Applications**
	Status: Gazette Notification sent through 2(14)/2014-EG-II Dated: 11th May, 2015 Subject: “Policy On Collaborative Application Development by Opening the Source Code of Government Applications” http://deity.gov.in/sites/upload_files/dit/files/policy_government_application.pdf
8	**Chapter 8: Application Development & Re-Engineering Guidelines for Cloud Ready Applications**
	Subject: Guidelines approved by Secretary through 2(8)/2013–EG-II Dated: 27th October ‘14 Title: “**Software Development & Re-Engineering Guidelines for Cloud Ready Applications**” http://deity.gov.in/sites/upload_files/dit/files/Application_Development_Re-Engineering_Guidelines.pdf

Digital India
Power To Empower

सत्यमेव जयते

Department of Electronics and Information Technology
Ministry of Communications and Information Technology
Government of India

E-Mail - secretary@deity.gov.in/asegov@deity.gov.in/jsegov@deity.gov.in
Phone : 011 - 24363075
Electronics Niketan, 6 CGO Complex, Lodhi Road, New Delhi 110003