



Open Source Good Governance Manuale

Autori: OSPO Alliance e partecipanti al GGI

Versione: v1.1

Data: 2023-09-04

Indice

1	Introduzione	3
1.1	Contesto	3
1.2	Informazioni riguardo la Good Governance Initiative	4
1.3	Informazioni riguardo la OSPO Alliance	4
1.4	Traduzioni	5
1.5	Contributori	5
1.6	Licenza	5
2	Organizzazione	7
2.1	Terminologia	7
2.2	Obiettivi	7
2.3	Attività canoniche	8
2.4	Customised Activity Scorecards	8
3	Metodologia	9
3.1	Preparare il terreno	9
3.2	Workflow	9
3.3	Setup manuale: utilizzo delle CAS (Customised Activity Scorecards)	10
3.4	Configurazione automatica: utilizzo della funzione GGI Deployment	11
3.5	Buon proseguimento	12
4	Attività dell'Obiettivo Utilizzo (Usage Goal)	13
4.1	Inventario delle competenze e delle risorse open source	13
4.2	Crescita delle competenze open source	14
4.3	Supervisione open source	16
4.4	Software aziendale open source	17
4.5	Gestire le competenze e le risorse open source	18
5	Attività dell'obiettivo fiducia (Trust Goal)	20
5.1	Gestire la conformità legale	20
5.2	Gestire le vulnerabilità del software	21
5.3	Gestire le dipendenze del software	23
5.4	Gestire gli indicatori chiave	25
5.5	Eseguire revisioni del codice	26
6	Attività dell'obiettivo culturale (Culture Goal)	28
6.1	Promuovere le migliori pratiche di sviluppo open source	28
6.2	Contribuire a progetti open source	29
6.3	Appartenere alla comunità open source	30
6.4	Prospettiva delle risorse umane	31
6.5	Upstream first	33
7	Attività dell'obiettivo di coinvolgimento (Engagement Goal)	35
7.1	Impegnarsi in progetti open source	35
7.2	Sostenere le comunità open source	36
7.3	Dichiarare pubblicamente l'uso dell'open source	37
7.4	Impegnarsi con i fornitori open source	37
7.5	Procurement policy open source	38
8	Attività dell'obiettivo strategico (Strategy Goal)	41
8.1	Impostare una strategia per la governance aziendale dell'open source	41
8.2	Consapevolezza dei C-Level	42
8.3	Open source e sovranità digitale	43
8.4	L'open source che consente l'innovazione	44
8.5	L'open source che consente la trasformazione digitale	46
9	Conclusioni	48
9.1	Contatti	48
9.2	Appendice: Modello di Customised Activity Scorecard	48

1 Introduzione

Questo documento introduce una metodologia per implementare una gestione professionale del software open source in un'organizzazione. Il documento affronta la necessità di utilizzare il software open source in modo corretto ed equo, di salvaguardare l'azienda da minacce tecniche, legali e di proprietà intellettuale e di massimizzare i vantaggi dell'open source. Qualunque sia la posizione di un'organizzazione su questi temi, questo documento propone indicazioni e idee per andare avanti e rendere il vostro viaggio un successo.

1.1 Contesto

La maggior parte degli utilizzatori finali e degli integratori di sistemi utilizza già codice libero e aperto (FOSS: Free e Open Source Software), sia nei propri sistemi informativi che all'interno delle proprie divisioni di prodotto o di servizi. La conformità al corretto utilizzo di software open source è una questione la cui importanza è cresciuta nel tempo, e molte grandi aziende hanno predisposto delle strutture dedicate alla conformità open source. Tuttavia, mentre è fondamentale impostare correttamente la filiera produttiva open source di un'azienda, attività che è poi proprio l'ambito di cui si occupa la conformità, è *necessario* che gli utilizzatori finali forniscano in cambio il proprio contributo alla sostenibilità dell'intero ecosistema open source. Noi consideriamo la buona governance dell'open source come un qualcosa che riguarda l'intero ecosistema, stabilendo legami con le comunità open source, promuovendo e alimentando una sana relazione con chi offre prodotti open source e servizi specialistici ad esso legati. Tutto questo porta la conformità ad un livello più elevato, e questo è ciò di cui si occupa la Good Governance Initiative.

Questa iniziativa va oltre le questioni di conformità e responsabilità. Essa riguarda la costruzione di una nuova consapevolezza nelle comunità di utilizzatori finali (spesso costituite esse stesse da sviluppatori) e integratori di sistemi, e riguarda lo sviluppo di relazioni mutuamente benefiche all'interno dell'ecosistema open source europeo.

Una Buona Governance dell'Open Source mette in grado organizzazioni di tutti i tipi (aziende piccole e grandi, amministrazioni civiche, università, associazioni, eccetera) di massimizzare i benefici derivanti dall'open source, aiutandole ad allineare persone, processi, tecnologie e strategie. E in quest'area di massimizzazione dei vantaggi del software open source, specialmente in Europa, chiunque può ancora imparare e innovare, senza che nessuno debba necessariamente sapere dove collocarsi riguardo allo stato dell'arte del settore.

Questa iniziativa ha l'obiettivo di aiutare le organizzazioni a raggiungere questi obiettivi tramite:

- Un catalogo strutturato di **attività**, una tabella di marcia per lo sviluppo di un approccio professionale alla gestione di software open source.
- Uno **strumento di gestione** per stabilire, controllare, riportare e comunicare tutte le informazioni relative ai progressi effettuati.
- Un **percorso chiaro e fattibile di progresso continuo**, eseguito a piccoli passi per ridurre i rischi, formare le persone, adattare i processi, comunicare internamente ed esternamente all'ambito organizzativo.
- **Linee guida** e un insieme di **riferimenti selezionati** circa licenze open source, buone pratiche, formazione e coinvolgimento nell'ecosistema per far leva su consapevolezza e cultura open source, consolidare la conoscenza interna ed estendere la direzione.

Questa guida è stata realizzata avendo in mente questi requisiti:

- Adottabilità del modello da parte di qualsiasi tipo di organizzazione: dalle piccole e medie imprese alle grandi aziende e alle organizzazioni senza scopo di lucro, dalle amministrazioni locali (ad esempio i Comuni) alle grandi istituzioni (ad esempio le istituzioni europee o i governi). Il modello fornisce gli elementi costitutivi per predisporre una strategia, assieme ai suggerimenti per la sua realizzazione, ma *come* le attività debbano essere svolte, dipende interamente dal contesto del programma di adozione e dal responsabile del programma stesso. Può risultare utile avvalersi di servizi di consulenza specifica e scambiare esperienze e pareri con altre organizzazioni che hanno intrapreso questo percorso.
- Non sono richiesti prerequisiti sul livello di conoscenza tecnica presente all'interno dell'organizzazione o dell'ambito di attività. Per esempio, alcune organizzazioni potranno

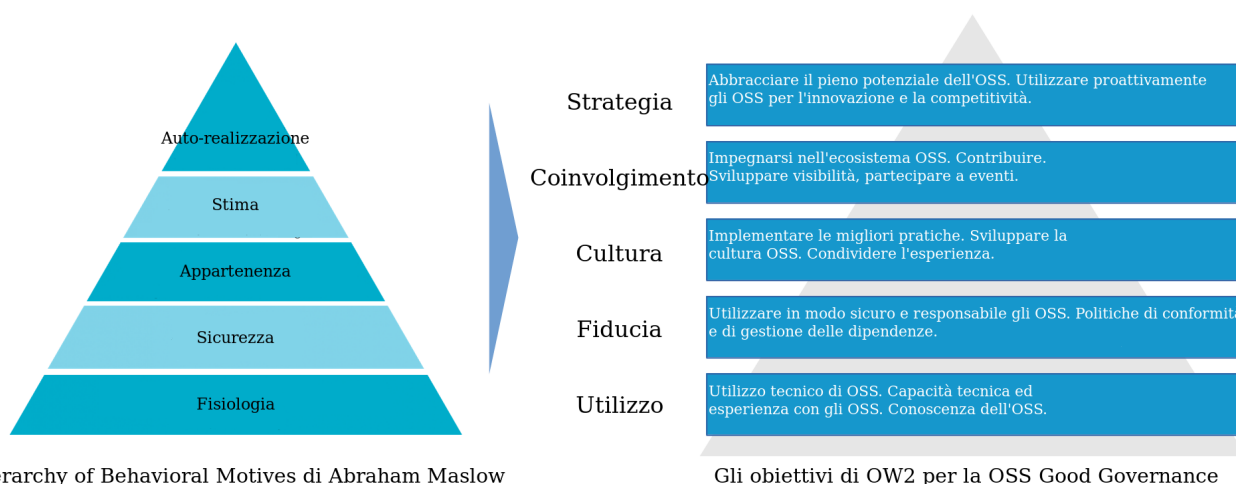
avere bisogno di un percorso completo di formazione, mentre altre potranno semplicemente fornire del materiale ad hoc ai diversi gruppi.

Alcune attività potranno non essere adatte a tutte le situazioni, ma l'intero modello fornisce una tabella di marcia e apre la strada a strategie per un suo adattamento o personalizzazione al contesto specifico.

1.2 Informazioni riguardo la Good Governance Initiative

In OW2, un'iniziativa rappresenta un impegno congiunto per soddisfare una specifica esigenza di mercato. La **Good Governance Initiative** propone un modello metodologico per sviluppare all'interno delle organizzazioni una modalità di gestione professionale dell'open source.

La Good Governance Initiative è basata su un modello esaustivo ispirato dalla gerarchia dei bisogni e delle motivazioni umane di Abraham Maslow, rappresentata nella figura seguente.



Attraverso idee, linee guida e attività, la Good Governance Initiative fornisce una specifica per la predisposizione di unità organizzative incaricate di gestire professionalmente il software open source. Queste unità organizzative sono talvolta indicate anche come OSPO, una sigla che sta per Open Source Program Office (un ufficio amministrativo espressamente dedicato alla gestione dell'open source). La metodologia è anche un sistema per definire le priorità, tenere sotto controllo e condividere i progressi dell'adozione del modello.

Man mano che progrediranno nell'adozione della metodologia GGI, le organizzazioni vedranno crescere le loro competenze in diversi ambiti, tra cui:

- l'**utilizzo** appropriato e sicuro del software open source all'interno dell'organizzazione, con il miglioramento del riuso, della manutenibilità e della rapidità nello sviluppo del software;
- la **mitigazione** dei rischi legali e tecnici associati all'utilizzo di codice e collaborazioni esterne all'organizzazione;
- l'**identificazione** della formazione necessaria ai diversi gruppi, dai singoli sviluppatori ai responsabili e ai dirigenti, in modo che tutti condividano la stessa visione;
- la capacità di stabilire **priorità** di obiettivi e attività, per sviluppare una strategia efficiente riguardo al software open source;
- la capacità di **comunicare** in maniera efficiente all'interno dell'organizzazione e verso l'esterno per sfruttare al meglio la strategia open source;
- il **miglioramento** della competitività dell'organizzazione e la sua capacità di attrarre i migliori talenti open source.

1.3 Informazioni riguardo la OSPO Alliance

La **OSPO Alliance** è stata avviata da una coalizione delle principali organizzazioni europee senza scopo di lucro, tra cui OW2, Eclipse Foundation, OpenForum Europe, e la Foundation for Public Code, avendo quale missione quella di far crescere la consapevolezza dell'open source in Europa e nel mondo, e di promuovere la gestione strutturata e professionale dell'open source da parte di aziende e amministrazioni.

Laddove la Good Governance Initiative si concentra sullo sviluppo di una metodologia di gestione, la OSPO Alliance ha l'obiettivo di più ampio respiro di aiutare le aziende, particolarmente nei settori non tecnologici, e le istituzioni pubbliche di conoscere e comprendere l'open source, iniziando a trarne beneficio nelle loro attività e a costituire al loro interno uffici amministrativi dedicati alla gestione dell'open source (OSPO: Open Source Program Office).

La OSPO Alliance ha predisposto il sito ufficiale **OSPO Alliance**, raggiungibile all'indirizzo <https://ospo-alliance.org>. La OSPO Alliance è al servizio della comunità offrendo un contesto affidabile per discutere e scambiare informazioni relative agli uffici per la gestione dell'open source (OSPO), e mettendo a disposizione un archivio contenente un insieme esaustivo di risorse per società, istituzioni pubbliche, organizzazioni accademiche e di ricerca. La OSPO Alliance consente di mettere in collegamento tutti i diversi uffici per la gestione dell'open source (OSPO) in Europa e nel mondo, così come quelle organizzazioni che a loro volta supportano le comunità. Incoraggia le migliori pratiche e promuove i contributi dedicati alla sostenibilità dell'open source. Consulta il sito [OSPO Alliance](https://ospo-alliance.org) per una veloce panoramica dei principali modelli relativi alle buone pratiche della gestione IT.

Il sito della [OSPO Alliance](https://ospo-alliance.org) è anche lo spazio in cui raccogliamo i riscontri riguardanti l'iniziativa e i suoi contenuti (ad esempio le attività, il corpus dei materiali) provenienti in generale dalla comunità.

1.4 Traduzioni

La traduzione del Manuale della Good Governance Initiative è un lavoro continuo portato avanti dalla comunità. Poiché il lavoro di traduzione è in continua evoluzione, raccomandiamo di consultare il nostro sito ufficiale per una lista aggiornata delle traduzioni disponibili.

Consulta la sezione <https://hosted.weblate.org/projects/ospo-zone-ggi/#languages>

La traduzione del Manuale della Open Source Good Governance Initiative è effettuata facendo uso di [Weblate](https://weblate.org/), un progetto e una piattaforma open source che permette di ospitare gratuitamente progetti open source di traduzione. Desideriamo ringraziarli vivamente, così come desideriamo ringraziare tutti coloro che contribuiscono alla traduzione. Siete meravigliosi.

1.5 Contributori

Queste persone straordinarie hanno contribuito alla Good Governance Initiative:

- Frédéric Aatz (Microsoft France)
- Boris Baldassari (Castalia Solutions, Eclipse Foundation)
- Philippe Bareille (Città di Parigi)
- Gaël Blondelle (Eclipse Foundation)
- Vicky Brasseur (Wipro)
- Philippe Carré (Nokia)
- Pierre-Yves Gibello (OW2)
- Michael Jaeger (Siemens)
- Sébastien Lejeune (Thales)
- Max Mehl (Free Software Foundation Europe)
- Hervé Pacault (Orange)
- Stefano Pampaloni (RIOS)
- Christian Paterson (OpenUp)
- Simon Phipps (Meshed Insights)
- Silvério Santos (Orange Business Services)
- Cédric Thomas (OW2)
- Nicolas Toussaint (Orange Business Services)

1.6 Licenza

Questo documento è rilasciato con licenza [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) (CC-BY 4.0). Dal sito Creative Commons:

Hai la libertà di:

- Condividerlo - copiare e redistribuire il materiale su qualunque supporto e in qualunque formato
- Adattarlo - usare il materiale per mischiare, trasformare e costruire nuovi contenuti a partire da esso

per qualsiasi scopo, anche commerciale.

Nel fornire la corretta paternità del contenuto, fornisci un collegamento alla licenza, indicando le eventuali modifiche che hai apportato. Puoi farlo in qualunque ragionevole maniera, purché questo non dia l'idea che il licenziante approvi te o il tuo utilizzo.

Tutti i diritti riservati: 2020-2022 OW2 e i partecipanti alla Open Source Good Governance Initiative.

2 Organizzazione

2.1 Terminologia

Il progetto della metodologia OSS Good Governance è strutturato attorno a quattro concetti chiave: Obiettivi, Attività canoniche, Scorecard di attività personalizzate e Iterazione.

- **Obiettivi (Goals):** Un obiettivo è un insieme di attività associate a un'area di interesse comune; esistono cinque obiettivi: Obiettivo Utilizzo, Obiettivo Fiducia, Obiettivo Cultura, Obiettivo Coinvolgimento e Obiettivo Strategia. Gli obiettivi possono essere raggiunti indipendentemente, in parallelo e perfezionati in modo iterativo attraverso le attività.
- **Attività canoniche (Canonical Activities):** all'interno di un obiettivo, un'attività affronta un singolo problema o argomento di sviluppo, come la gestione della conformità legale, che può essere utilizzato come passo incrementale verso gli obiettivi del programma. L'insieme completo delle attività definite dal GGI è chiamato Attività canoniche.
- **Scorecard di attività personalizzate (CAS - Customised Activity Scorecard):** per implementare il GGI in una determinata organizzazione, le attività canoniche devono essere adattate alle specificità del contesto, costruendo così una serie di schede di valutazione delle attività personalizzate. La scheda di valutazione dell'attività personalizzata descrive come l'attività verrà implementata nel contesto dell'organizzazione e come verranno monitorati i progressi.
- **Iterazione (Iteration):** L'OSS Good Governance è un sistema di gestione e come tale richiede una valutazione, un riesame e una revisione periodica. Pensate al sistema contabile di un'organizzazione, è un processo continuo con almeno un punto di controllo annuale, il bilancio; allo stesso modo, il processo di buona governance dell'OSS richiede almeno una revisione annuale, ma le revisioni possono essere parziali o più frequenti a seconda delle attività.

2.2 Obiettivi

Le Attività canoniche definite dal GGI sono organizzate in obiettivi. Ogni obiettivo affronta un'area specifica di progresso all'interno del processo. Dall'Uso alla Strategia, gli Obiettivi coprono questioni relative a tutte le parti interessate, dai team di sviluppo fino al management di livello superiore.

- **Obiettivo Utilizzo (Usage Goal):** questo obiettivo riguarda le fasi preliminari dell'utilizzo del software open source. Le attività relative all'Obiettivo Utilizzo coprono i primi passi di un programma open source, identificando l'efficienza con cui l'open source viene utilizzato e i vantaggi che apporta all'organizzazione. Include la formazione e la gestione delle conoscenze, la produzione di inventari dell'open source già utilizzato all'interno dell'azienda e la presentazione di alcuni concetti di open source che possono essere utilizzati nel corso del processo.
- **Obiettivo Fiducia (Trust Goal):** questo obiettivo riguarda l'utilizzo sicuro dell'open source. Trust Goal si occupa della conformità legale, della gestione delle dipendenze e delle vulnerabilità e, in generale, mira a creare fiducia nel modo in cui l'organizzazione utilizza e gestisce l'open source.
- **Obiettivo Culturale (Culture Goal):** l'obiettivo culturale comprende attività volte a mettere i team a proprio agio con l'open source, a partecipare individualmente alle attività di collaborazione, a comprendere e a implementare le migliori pratiche dell'open source. Questo obiettivo promuove un senso di appartenenza alla comunità open source tra gli individui.
- **Obiettivo Coinvolgimento (Engagement Goal):** Questo obiettivo mira a impegnare un'organizzazione nell'ecosistema open source. Vengono stanziati risorse umane e finanziarie per contribuire ai progetti open source. In questo caso, l'organizzazione assume il proprio ruolo civico nell'open source e la responsabilità di garantire la sostenibilità dell'ecosistema.
- **Obiettivo Strategia (Strategy goal):** questo obiettivo consiste nel rendere l'open source visibile e accettabile ai livelli più alti della gestione aziendale. Si tratta di riconoscere che l'open source è un fattore strategico di sovranità digitale, di innovazione dei processi e, in generale, una fonte di attrattiva e di buona volontà.

2.3 Attività canoniche

Le Canonical Activities (attività canoniche) sono al centro del progetto GGI. Nella sua versione iniziale, la Metodologia GGI prevede cinque Canonical Activities per ogni obiettivo, per un totale di 25. Le Canonical Activities sono descritte utilizzando le seguenti sezioni predefinite:

- *Description*: una sintesi dell'argomento trattato dall'attività e delle fasi di completamento.
- *Opportunity Assessment*: descrive perché e quando è importante intraprendere questa attività.
- *Progress Assessment*: descrive come misurare i progressi dell'attività e valutarne il successo.
- *Tools*: un elenco di tecnologie o strumenti che possono aiutare a realizzare questa attività.
- *Recommendations*: suggerimenti e buone pratiche raccolte dai partecipanti al GGI.
- *Resources*: link e riferimenti per approfondire l'argomento trattato nell'attività.

Descrizione

Questa sezione fornisce una descrizione di alto livello delle Activity, una sintesi dell'argomento per definire lo scopo dell'attività nel contesto dell'approccio open source all'interno di un obiettivo.

Valutazione delle opportunità

Per aiutare a strutturare un approccio iterativo, ogni attività ha una sezione "Opportunity Assessment" (Valutazione dell'opportunità), con una o più domande allegate. La valutazione dell'opportunità si concentra sul motivo per cui è importante intraprendere l'attività e su quali esigenze essa risponde. La valutazione dell'opportunità aiuterà a definire gli sforzi previsti, le risorse necessarie e a valutare i costi e il ROI atteso.

Valutazione dei progressi

Questa fase si concentra sulla definizione di obiettivi, KPI e sulla fornitura di *punti di verifica* che aiutino a valutare i progressi dell'attività. I punti di verifica sono suggeriti, ma possono aiutare a definire una tabella di marcia per il processo di buona governance, le sue priorità e le modalità di misurazione dei progressi.

Strumenti

Di seguito sono elencati gli strumenti che possono aiutare a svolgere l'attività o a strumentare una fase specifica dell'attività. Gli strumenti non sono una raccomandazione obbligatoria, né pretendono di essere esaustivi, ma sono suggerimenti o categorie da elaborare in base al contesto esistente.

Raccomandazioni

Questa sezione viene regolarmente aggiornata con i feedback degli utenti e con tutti i tipi di raccomandazioni che possono aiutare a gestire l'attività.

Risorse

Vengono proposte risorse per alimentare l'approccio con studi di base, documenti di riferimento, eventi o contenuti online per arricchire e sviluppare l'approccio relativo all'attività. Le risorse non sono esaustive, sono punti di partenza o suggerimenti per ampliare la semantica dell'attività in base al proprio contesto.

2.4 Customised Activity Scorecards

Le Customised Activity Scorecards (CAS) sono leggermente più dettagliate delle attività canoniche. Un CAS include dettagli specifici dell'organizzazione che implementa il GGI. L'uso delle CAS è descritto nella sezione Metodologia.

3 Metodologia

L'implementazione della metodologia di Good Governance OSS è in definitiva un'iniziativa di forte impatto. Coinvolge diverse categorie di persone, servizi e processi aziendali, dalle pratiche quotidiane alla gestione delle risorse umane, dagli sviluppatori ai dirigenti di alto livello. Per implementare una buona governance dell'open source non esiste un meccanismo universale. Diversi tipi di organizzazione, diverse culture e situazioni aziendali richiederanno approcci diversi alla governance dell'open source. Per ogni organizzazione ci saranno vincoli e aspettative diverse, che porteranno a percorsi e modi diversi di gestione del programma.

In queste circostanze, la metodologia GGI fornisce un progetto generico che può essere adattato al settore, alla cultura e alle esigenze specifiche di un'organizzazione. Sebbene il progetto sia inteso come completo e autosufficiente, la metodologia può essere implementata progressivamente. È possibile avviare il programma selezionando gli obiettivi e le attività ritenuti rilevanti in un contesto specifico: l'idea è quella di costruire una prima bozza di roadmap, che aiuti a dare il via all'iniziativa.

Oltre a questo quadro di riferimento, consigliamo vivamente di entrare in contatto con i colleghi attraverso una rete consolidata come l'iniziativa europea [OSPO Alliance](#), o altre iniziative simili del gruppo TODO o OSPO++. L'importante è poter scambiare feedback con persone che gestiscono un'iniziativa simile e condividere i problemi incontrati e le soluzioni esistenti.

3.1 Preparare il terreno

Data l'ambizione della metodologia GGI e il suo impatto potenzialmente ampio, è importante comunicare con diverse persone all'interno di un'organizzazione. Sarebbe opportuno coinvolgerli per stabilire una serie iniziale di aspettative e requisiti realistici per iniziare con il piede giusto, attirare l'interesse e il sostegno. Una buona idea è quella di pubblicare le Customised Activity Scorecards (CAS) sulla piattaforma collaborativa dell'organizzazione, in modo che possano essere utilizzate per comunicare con le parti interessate. Alcuni suggerimenti:

- Identificate le principali parti interessate e fate in modo che si accordino su una serie di obiettivi primari. Coinvolgeteli nel successo dell'iniziativa come parte del loro programma.
- Ottenete il consenso iniziale, concordate le fasi e il ritmo e stabilite controlli regolari per informarli dei progressi.
- Assicuratevi che comprendano i vantaggi di ciò che si può ottenere e di ciò che comporta: i miglioramenti attesi devono essere chiari e i risultati visibili.
- Stabilite una prima diagnosi o lo stato dell'arte dell'open source nell'organizzazione candidata. Il risultato: un documento che descrive i risultati di questo programma, la situazione dell'organizzazione e la direzione che intende prendere.

3.2 Workflow

Come moderni professionisti del software, ci piacciono i metodi agili che definiscono miglioramenti piccoli e sicuri, poiché è buona norma rivalutare regolarmente la situazione e fornire risultati intermedi significativi.

Questo è molto importante nel contesto di un programma OSPO in corso, poiché molti aspetti cambieranno nel tempo, dalla strategia e dalla risposta dell'organizzazione all'open source alla disponibilità e all'impegno delle persone. La rivalutazione e l'iterazione periodica consentono inoltre di adattarsi all'accettazione del programma in corso, di monitorare meglio le tendenze e le opportunità attuali e di ottenere piccoli vantaggi incrementali per gli stakeholder e per l'organizzazione nel suo complesso.

Idealmente, la metodologia potrebbe essere implementata in cinque fasi come segue:

1. **Discovery** Comprendere i concetti chiave, appropriarsi della metodologia, allineare gli obiettivi e le aspettative.
2. **Customisation** Adattare la descrizione dell'attività e la valutazione delle opportunità alle specificità dell'organizzazione.
3. **Prioritisation** Identificare gli obiettivi e i risultati chiave, i compiti e gli strumenti, programmare le tappe e redigere la tempistica.

4. **Activation** Finalizzazione di Scorecard, budget, incarichi, documentazione dei compiti su issue manager.
5. **Iteration** Valutare e raccogliere i risultati, evidenziare i problemi, migliorare, aggiustare. Iterazione ogni trimestre o semestre.

Preparazione della prima iterazione del programma:

- Identificare una prima serie di compiti su cui lavorare e stabilire una priorità in base alle esigenze (i passaggi che separano dallo stato desiderato) e alla tempistica. Risultato: un elenco di compiti su cui lavorare durante l'iterazione.
- Definire una serie di requisiti e aree di miglioramento, comunicarli alle parti interessate e agli utenti finali e ottenere la loro approvazione o il loro impegno.
- Compilare le schede di valutazione per monitorare i progressi. Un modello di scorecard può essere scaricato dal [repository GGI](#).

Alla fine di ogni iterazione, fate una retrospettiva e preparatevi per l'iterazione successiva:

- Comunicare gli ultimi miglioramenti.
- Valutate a che punto siete, se i compiti mirati sono stati completati, perfezionate la roadmap di conseguenza.
- Verificate i punti dolenti o i problemi rimanenti, chiedete il supporto di altri responsabili o servizi se necessario.
- Ridefinire le priorità dei compiti in base al contesto aggiornato.
- Definire un nuovo sottoinsieme di compiti da eseguire.

3.3 Setup manuale: utilizzo delle CAS (Customised Activity Scorecards)

Una CAS è un modulo che descrive una Canonical Activity personalizzata in base alle specificità di un'organizzazione. L'insieme delle schede di valutazione delle attività personalizzate fornisce la tabella di marcia per la gestione del software open source.

Si noti che, in base alle prime esperienze con la metodologia, è necessaria fino a un'ora per adattare una Canonical Activity alla relativa Score Card personalizzata di un'organizzazione.

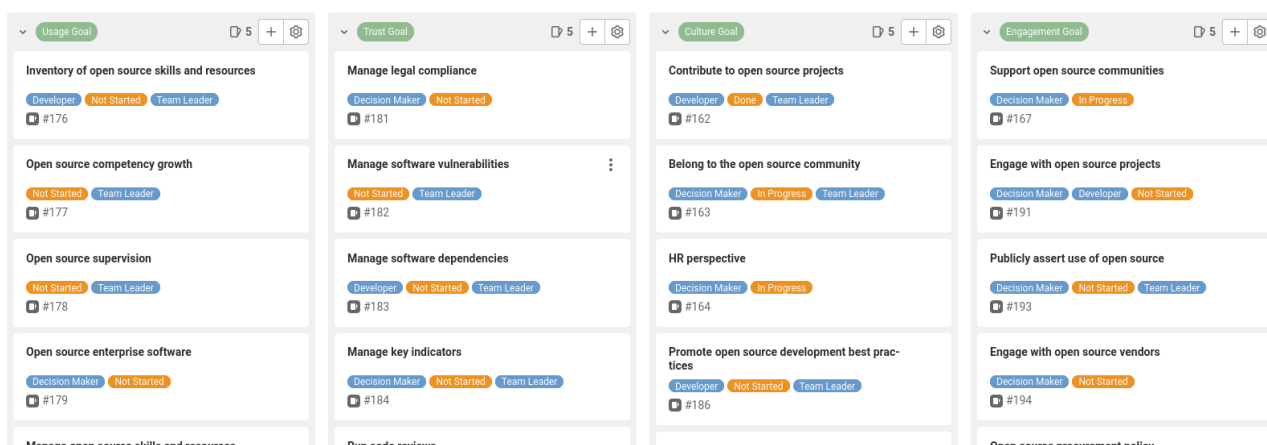
Le Customised Activity Scorecard contengono le seguenti sezioni:

- **Title Disambiguation** Prima di tutto, prendete qualche minuto per capire di cosa potrebbe trattarsi l'attività e la sua rilevanza, e come può inserirsi nel vostro percorso complessivo di gestione degli OSS.
- **Customised Description** Adattare l'attività alle specificità dell'organizzazione. Definite l'ambito dell'attività, il caso d'uso specifico che affronterete.
- **Opportunity Assessment** Spiegate perché è importante intraprendere questa attività, quali esigenze soddisfa. Quali sono i nostri punti dolenti? Quali sono le opportunità di progresso? Cosa si può ottenere?
- **Objectives** Definire un paio di obiettivi cruciali per l'attività. Punti dolenti da risolvere, opportunità di progresso, desideri. Identificare i compiti chiave. Ciò che si vuole ottenere in questa iterazione.
- **Tools** Tecnologie, strumenti e prodotti utilizzati nelle Activity.
- **Operational Notes** Indicazioni sull'approccio, il metodo e la strategia da seguire in questa Activity.
- **Key Results** Definire risultati attesi misurabili e verificabili. Scegliere i risultati che indicano i progressi rispetto agli Obiettivi. Indicare qui i KPI.
- **Progress and Score** Progress è, in %, la percentuale di completamento del risultato; Score è la valutazione del successo personale.
- **Personal Assessment** Per ogni risultato è possibile aggiungere una breve spiegazione e spiegare il proprio grado di soddisfazione personale espresso nel punteggio.
- **Timeline** Indicare le date di inizio-fine (Start-End dates), le fasi di lavoro (Phasing tasks), i passaggi critici (Critical steps), le tappe (Milestones).
- **Efforts** Valutare le risorse di tempo e materiali richieste, interne e di terzi. Quali sono gli sforzi previsti? Quanto costerà? Di quali risorse abbiamo bisogno?
- **Assignees** Dire chi partecipa. Assegnate i compiti o la leadership e le responsabilità dell'Activity.

- **Issues** Identificare i problemi chiave, le difficoltà previste, i rischi, gli ostacoli, le incertezze, i punti di attenzione, le dipendenze critiche.
- **Status** Scrivere qui una valutazione sintetica di come sta l'attività: in salute? In ritardo? ecc.
- **Overall Progress Rating** La vostra valutazione sintetica dei progressi dell'Activity, di alto livello e orientata alla gestione.

3.4 Configurazione automatica: utilizzo della funzione GGI Deployment

A partire dalla versione Handbook 1.1, il GGI propone **My GGI Board**, uno strumento automatizzato per distribuire la propria istanza del GGI come progetto GitLab. Il processo di installazione richiede meno di 10 minuti, è completamente documentato e fornisce un modo semplice e affidabile per personalizzare le attività, seguirne l'esecuzione man mano che si compiono progressi e comunicare i risultati alle parti interessate. Un esempio dal vivo della distribuzione può essere visto nella [initiative's GitLab](#), con il sito web generato automaticamente disponibile sulle [relative GitLab Pages](#).



Ecco un flusso di lavoro standard per utilizzare la funzione di deployment:

1. Effettuare un fork della scheda My GGI alla propria istanza o progetto GitLab e configurarla seguendo le istruzioni contenute nel README del progetto: <https://gitlab.ow2.org/ggi/my-ggi-board>. In questo modo:
 - Creare tutte le attività come issue nel progetto.
 - Creare una tabella dall'aspetto accurato per aiutarvi a visualizzare e gestire le attività.
 - Creare un sito web statico, servito dalle pagine dell'istanza GitLab, con le informazioni estratte dalle attività.
 - Aggiornare la descrizione del progetto con i link corretti che puntano alla board delle attività e al vostro sito web statico.
1. Da qui si può iniziare a esaminare le diverse attività e a compilare la sezione della scorecard.
 - La sezione delle schede di valutazione è l'equivalente elettronico (e semplificato) delle schede di valutazione ODT di cui sopra. Si usa per adattare l'attività al proprio contesto, elencando le risorse locali, i rischi e le opportunità e definendo gli obiettivi personalizzati necessari per completare l'attività.
 - Se un'attività non si applica al vostro contesto, contrassegnatela semplicemente come "Not Selected" o chiudetela.
 - Si tratta di un processo che richiede molto tempo, ma che è assolutamente necessario perché vi aiuterà, passo dopo passo, a definire la vostra tabella di marcia e il vostro piano.
1. Una volta definite le attività, è possibile iniziare a implementare il proprio OSPO. Selezionate alcune attività che ritenete rilevanti per iniziare e modificate la loro etichetta di avanzamento da "Not Started" a "In Progress". Potete utilizzare le funzioni di GitLab per organizzare il lavoro (commenti, assegnatari, ecc.) o qualsiasi altro strumento. È facile collegare le attività e sono disponibili molte integrazioni.
2. A intervalli regolari (settimanali, mensili, a seconda del vostro calendario), valutate e rivedete le attività in corso e, una volta completate, cambiate l'etichetta da "In Progress" a "Done". Selezionatene altre e ricominciate dal punto 3 finché non saranno tutte completate.

Il sito web propone una rapida panoramica delle attività attuali e passate ed estrae la sezione scorecard dei temi per visualizzare solo le informazioni rilevanti a livello locale. Quando si verificano cambiamenti nelle issue (attività), questi vengono aggiornati automaticamente nel sito web generato. Si noti che le pipeline CI per la generazione automatica del sito web vengono eseguite automaticamente di notte, ma è possibile lanciarle facilmente dalla sezione CI/CD del progetto GitLab. L'immagine seguente mostra l'interfaccia del sito web generato automaticamente.

My Good Governance Initiative Dashboard My Board

Welcome

This is the website of your own good governance Initiative.

There are currently:

- 17 activities `not_started`
- 4 activities `in_progress`
- 4 activities `done`

Current activities [\[details \]](#)

Current activities are defined as having the label `in_progress`.

These are your current activities:

- [Open source enabling digital transformation](#) (GGI-A-37).
Tasks: 1 done / 2 total.
50%
- [Support open source communities](#) (GGI-A-30).
Tasks: 2 done / 3 total.
66%

È possibile porre domande o ottenere supporto per la funzione di distribuzione sulla nostra homepage di GitLab e siamo lieti di ricevere feedback.

Homepage di GGI Deploy: <https://gitlab.ow2.org/ggi/my-ggi-board>

3.5 Buon proseguimento

Comunicare il vostro successo e godetevi la tranquillità che deriva da una strategia open source all'avanguardia!

L'OSS Good Governance è un metodo per attuare un programma di miglioramento continuo, e come tale non finisce mai. Tuttavia, è importante evidenziare le fasi intermedie e apprezzare i cambiamenti che produce, per rendere visibili i progressi e condividere i risultati.

- Comunicare con gli stakeholder e gli utenti finali per far conoscere loro i vantaggi e i benefici che lo sforzo dell'iniziativa comporta.
- Promuovere la sostenibilità del programma. Assicurare che le migliori pratiche e le lezioni apprese dal programma siano sempre applicate e aggiornate.
- Condividete la vostra esperienza con i vostri colleghi: fornite un feedback al gruppo di lavoro GGI e alla vostra comunità di adozione OSPO e condividete il vostro approccio.

4 Attività dell'Obiettivo Utilizzo (Usage Goal)

4.1 Inventario delle competenze e delle risorse open source

ID Activity: [GGI-A-17](#).

Descrizione

In qualsiasi fase, dal punto di vista della gestione, è utile avere una mappatura, un inventario delle risorse open source, degli asset, dell'utilizzo e del loro stato, nonché delle esigenze potenziali e delle soluzioni disponibili. Ciò include anche la valutazione dell'impegno e delle competenze necessarie per colmare i gap.

Questa attività mira a fare un'istantanea della situazione dell'open source all'interno dell'organizzazione e sul mercato e a valutare il ponte tra loro.

- Inventario dell'utilizzo di OSS nella catena di sviluppo del software e nei prodotti e componenti software utilizzati in produzione.
- Identificare le tecnologie open source (soluzioni, framework, funzioni innovative) che potrebbero soddisfare le vostre esigenze e contribuire a migliorare il vostro processo.

Non incluso

- Identificare e qualificare gli ecosistemi e le comunità OSS correlate. (Culture Goal)
- Identificare le dipendenze da librerie e componenti OSS. (Trust Goal)
- Identificare le competenze tecniche (ad es. linguaggi, framework...) e soft (ad es. collaborazione, comunicazione) necessarie. (appartiene alle successive Activity: Crescita delle competenze OSS e Competenze per lo sviluppo di software open source)

Valutazione delle opportunità

Un inventario delle risorse open source disponibili che aiuterà a ottimizzare gli investimenti e a dare priorità allo sviluppo delle competenze.

Questa attività crea le condizioni per migliorare la produttività dello sviluppo, data l'efficienza e la popolarità dei componenti, dei principi di sviluppo e degli strumenti OSS, in particolare nello sviluppo di applicazioni e infrastrutture moderne.

- Potrebbe essere necessario semplificare il portfolio di risorse OSS.
- Potrebbe essere necessario riqualificare il personale.
- Ciò consente di identificare le esigenze e di alimentare la roadmap IT.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste un elenco praticabile di risorse OSS "We use", "We integrate", "We produce", "We host", e le relative Competenze
- Siamo impegnati a migliorare l'efficienza utilizzando metodi e strumenti all'avanguardia.
- Abbiamo identificato risorse OSS che non sono ancora state elencate (che potrebbero essere emerse internamente: è possibile definire una politica in questo settore?)
- Chiediamo ai nuovi progetti di appoggiare o riutilizzare le risorse OSS esistenti. (Obiettivo culturale?)
- Abbiamo una percezione e una comprensione ragionevolmente sicura dell'ambito di utilizzo degli OSS nella nostra organizzazione.

Strumenti

Ci sono molti modi diversi per stabilire tale inventario. Un modo potrebbe essere quello di classificare le risorse OSS in quattro categorie:

- OSS che utilizziamo: software che utilizziamo in produzione o in sviluppo
- OSS che integriamo: ad esempio, le librerie OSS che integriamo in un'applicazione personalizzata

- OSS che produciamo: ad esempio, una libreria che abbiamo pubblicato su GitHub o un progetto OSS che sviluppiamo o a cui contribuiamo regolarmente.
- OSS che ospitiamo: OSS che gestiamo per offrire un servizio interno come un CRM, GitLab, Nexus, ecc. Un esempio di tabella potrebbe essere il seguente:

Utilizziamo	Integriamo	Produciamo	Noi ospitiamo	Competenze
Firefox, OpenOffice, Postgresql	Biblioteca slf4j	Biblioteca YY su GH	GitLab, Nexus	Java, Python

La stessa identificazione dovrebbe valere per le competenze

- Competenze ed esperienze disponibili nei team esistenti
- Competenze ed esperienze che potrebbero essere sviluppate o acquisite internamente (formazione, coaching, esperimento)
- Competenze ed esperienze che devono essere ricercate sul mercato o attraverso partenariati/appalti

Raccomandazioni

- Mantieni le cose semplici.
- Si tratta di un esercizio di livello relativamente alto, non di un inventario dettagliato per il reparto contabilità.
- Sebbene questa attività sia un buon punto di partenza, non è necessario completarla al 100% prima di avviare le altre attività.
- Gestire le problematiche, le risorse e le competenze relative allo **sviluppo del software** nell'Activity #42.
- L'inventario deve coprire tutte le categorie IT: sistemi operativi, middleware, DBMS, amministrazione di sistema, strumenti di sviluppo e di test, ecc.
- Iniziate a identificare le comunità affini: è più facile ottenere supporto e feedback dal progetto quando vi conoscono già.

Risorse

- Un eccellente corso su [Free \(/Libre\), and Open Source Software \(FOSS\)](#), tenuto dal professor Dirk Riehle.

4.2 Crescita delle competenze open source

ID attività: [GGI-A-18](#).

Descrizione

Questa attività riguarda la pianificazione e l'avvio delle competenze tecniche e delle prime esperienze con gli OSS, una volta effettuato l'inventario (#17). È anche l'occasione per iniziare a costruire una tabella di marcia semplificata per lo sviluppo delle competenze.

- Identificare le competenze e la formazione necessarie.
- Creare un progetto pilota per dare il via all'approccio, imparare "facendo", identificare una prima milestone.
- Capitalizzare le lezioni apprese e costruire un insieme di conoscenze.
- Iniziare a identificare e documentare i passi successivi per un'adozione più ampia.
- Elaborare una strategia nei prossimi mesi o un anno per ottenere il supporto finanziario e gestionale.

L'ambito dell'attività:

- Linux, Apache, Debian, competenze amministrative.
- Database open source MariaDB, MySQL, PostgreSQL, ecc.
- Virtualizzazione open source e tecnologie cloud.
- Lo stack LAMP e le sue alternative.

Valutazione delle opportunità

Come ogni tecnologia IT, e probabilmente anche di più, l'open source porta innovazione. L'open source cresce velocemente e cambia rapidamente. Richiede alle organizzazioni di tenersi aggiornate.

Questa attività aiuta a identificare le aree in cui la formazione potrebbe aiutare le persone a diventare più efficienti e a sentirsi più sicure nell'uso dell'open source. Aiuta a prendere decisioni sullo sviluppo dei dipendenti. L'acquisizione di competenze di base sull'open source consente di valutare l'opportunità di:

- Estendere le soluzioni IT con le tecnologie di mercato esistenti sviluppate dall'ecosistema.
- Sviluppare nuove modalità di collaborazione all'interno e all'esterno dell'organizzazione.
- Acquisire competenze in tecnologie nuove e innovative.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Viene sviluppata una matrice di competenze.
- L'ambito delle tecnologie OSS utilizzate è definito in modo proattivo, evitando cioè l'uso incontrollato di tecnologie OSS.
- Per queste tecnologie viene acquisito un livello di competenza soddisfacente.
- I team hanno ricevuto una formazione "open source Basics" per iniziare.

Strumenti

Uno strumento chiave in questo caso è la Matrice (o Mappatura) delle attività (o delle competenze).

Questa attività può essere svolta:

- utilizzando tutorial online (molti gratuitamente su Internet),
- partecipando a conferenze per sviluppatori,
- Attraverso la formazione professionale, ecc.

Raccomandazioni

- L'utilizzo e lo sviluppo di componenti open source in modo sicuro ed efficiente richiede una mentalità aperta e collaborativa che deve essere riconosciuta e diffusa sia dall'alto (management) che dal basso (sviluppatori).
- Assicurarsi che l'approccio sia attivamente sostenuto e promosso dalla direzione. Non si otterrà nulla senza il coinvolgimento del management.
- Coinvolgere le persone (sviluppatori, stakeholder) nel processo: organizzare tavole rotonde e ascoltare le idee.
- Lasciate che le persone abbiano tempo e risorse per scoprire, provare e giocare con questi nuovi concetti. Se possibile, rendetelo divertente: la gamification e i premi sono ottimi incentivi.

Un progetto pilota con le seguenti fasi potrebbe fungere da catalizzatore:

- Identificare la tecnologia o il framework con cui iniziare.
- Trovate formazione online, esercitazioni e codice di esempio per sperimentare.
- Costruire un prototipo della soluzione finale.
- Identificate alcuni esperti per sfidare e allenare l'implementazione.

Risorse

- [Cos'è una matrice di competenze](#): una rapida lettura introduttiva.
- [Come creare una matrice di competenze per il vostro team](#): un modello con commenti.
- [MOOC sulla cultura libera \(libre\)](#) (solo in francese): si tratta di un corso in 6 parti sulla cultura libera, con introduzione ai diritti d'autore, alla proprietà intellettuale e alle licenze open source

4.3 Supervisione open source

ID Activity: [GGI-A-19](#).

Descrizione

Questa attività riguarda il controllo dell'uso dell'open source e la garanzia che il software sia gestito in modo proattivo. Ciò riguarda diverse prospettive, sia che si tratti di utilizzare strumenti e soluzioni aziendali OSS, sia che si tratti di includere OSS come componenti nei propri sviluppi o di modificare una versione di un software adattandola alle proprie esigenze. Si tratta anche di identificare le aree in cui l'open source è diventato una soluzione di fatto (a volte nascosta) e di valutarne l'idoneità.

Potrebbe essere necessario chiarire quanto segue:

- Viene fornita la funzionalità richiesta?
- Vengono fornite funzionalità aggiuntive non necessarie, ma che aumentano la complessità delle fasi di BUILD e RUN?
- Cosa richiede la licenza, quali sono i vincoli legali?
- In che misura la decisione rende la vostra organizzazione indipendente dai fornitori?
- Esiste un'opzione di supporto, pronta per le vostre esigenze aziendali, e quanto costa?
- TCO (Total Cost of Ownership).
- Il management conosce i vantaggi dell'open source, ad esempio al di là del "risparmio sui costi di licenza"? Essere a proprio agio con l'open source aiuta a trarre il massimo beneficio dalla collaborazione con le comunità di progetto e i fornitori.
- Valutare se ha senso condividere i costi di sviluppo restituendo i propri sviluppi alla comunità, tenendo conto di tutte le implicazioni, come la conformità alla licenza.
- Verificare la disponibilità di un supporto comunitario o professionale.

Valutazione delle opportunità

Definire un processo decisionale specificamente rivolto all'open source è un modo per massimizzarne i vantaggi.

- In questo modo si evita l'emergere di usi incontrollati e di costi nascosti delle tecnologie open source.
- Porta a decisioni strategiche e organizzative informate e consapevoli degli OSS.

Costi: l'attività può mettere in discussione e riconsiderare l'uso de facto non ottimale dell'open source come inefficiente, rischioso, ecc.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- L'OSS è diventato un'opzione comoda quando la scelta dell'OSS non è vista come un'eccezione o una scelta pericolosa.
- L'OSS è diventato un'opzione "mainstream".
- Gli attori principali sono sufficientemente convinti che la soluzione open source offra vantaggi strategici che valgono la pena di essere investiti.
- È possibile dimostrare che il TCO della soluzione basata sull'open source offre alla vostra organizzazione un valore superiore rispetto all'alternativa.
- Viene valutato il modo in cui l'indipendenza dei fornitori fa risparmiare denaro o può potenzialmente farlo in futuro.
- È stato valutato che l'indipendenza della soluzione riduce i rischi di un cambiamento troppo costoso della soluzione (non sono possibili formati di dati chiusi).

Strumenti

In questa fase, non possiamo pensare ad alcuno strumento rilevante o interessato da questa attività.

Raccomandazioni

- Gestire in modo proattivo l'uso dell'open source richiede livelli base di consapevolezza e comprensione dei fondamenti dell'open source, che devono essere presi in considerazione in ogni decisione relativa all'OSS.
- Confrontate le funzionalità necessarie invece di cercare un'alternativa per una soluzione closed source conosciuta.
- Assicuratevi di avere supporto e ulteriori sviluppi.
- Considerate gli effetti della licenza della soluzione sulla vostra organizzazione.
- Convincere tutti gli attori chiave del valore dei vantaggi dell'open source, al di là del "risparmio sui costi di licenza".
- Siate onesti, non esagerando con i benefici attesi dall'open source.
- Nel processo decisionale è altrettanto importante valutare le diverse soluzioni open source per evitare delusioni dovute ad aspettative sbagliate, per chiarire cosa si richiede all'organizzazione e tutti i vantaggi che l'apertura delle soluzioni comporta. Questo deve essere identificato in modo che l'organizzazione possa valutarlo per il proprio contesto.

Risorse

- [Top 5 Benefits of Open Source](#): Blog sponsorizzato, ma comunque interessante e di rapida lettura.
- [Weighing The Hidden Costs Of Open Source](#): uno studio sponsorizzato da IBM sui costi di supporto dell'OSS.

4.4 Software aziendale open source

ID Activity: [GGI-A-20](#).

Descrizione

Questa attività riguarda la selezione proattiva di soluzioni OSS, sia di fornitori che di comunità, in aree orientate al business. Può anche riguardare la definizione di politiche di preferenza per la selezione di software applicativo aziendale open source.

Sebbene il software open source sia più spesso utilizzato dai professionisti dell'IT (sistema operativo, middleware, DBMS, amministrazione di sistema, strumenti di sviluppo), non è ancora stato riconosciuto nelle aree in cui i professionisti del business sono gli utenti principali.

L'attività riguarda aree quali: Suite per ufficio, ambienti di collaborazione, gestione degli utenti, gestione dei flussi di lavoro, gestione delle relazioni con i clienti, e-mail, e-Commerce, ecc.

Valutazione delle opportunità

Man mano che l'open source tende a generalizzarsi si estende ben oltre i sistemi operativi e gli strumenti di sviluppo, trovando sempre più spazio negli strati superiori dei sistemi informativi, fino alle applicazioni aziendali. È importante identificare quali applicazioni OSS vengono utilizzate con successo per soddisfare le esigenze dell'organizzazione e come possono diventare la scelta preferita di un'organizzazione per risparmiare sui costi.

L'attività può comportare alcuni costi di riqualificazione e migrazione.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste un elenco di soluzioni OSS consigliate per soddisfare le esigenze in sospeso nelle applicazioni aziendali.
- Viene redatta una politica di preferenza per la selezione di software applicativi aziendali open source.
- Le applicazioni aziendali proprietarie in uso vengono valutate rispetto agli equivalenti OSS.
- I processi di approvvigionamento e gli inviti a presentare proposte specificano la preferenza per l'open source (se legalmente possibile).

Strumenti

Strumenti per la mappatura di software e applicazioni aziendali?

Al momento non siamo a conoscenza di strumenti rilevanti per questa attività.

Raccomandazioni

- Parlate con i colleghi, imparate da ciò che fanno le altre aziende simili alla vostra.
- Visitate gli eventi di settore locali per scoprire le soluzioni OSS e il supporto professionale.
- Provate le edizioni della comunità e il supporto della comunità prima di impegnarvi in piani di supporto a pagamento.

Risorse

- [Cos'è l'open source aziendale?](#): una rapida lettura sull'open source pronto per le imprese.
- [101 applicazioni open source per aiutare la vostra azienda a prosperare](#): Un elenco indicativo di soluzioni open source orientate al business.

4.5 Gestire le competenze e le risorse open source

Activity ID: [GGI-A-42](#).

Descrizione

Questa attività è incentrata su competenze e risorse in **software development**. Include le tecnologie e le competenze specifiche degli sviluppatori, nonché i processi di sviluppo, i metodi e gli strumenti generali.

Per le tecnologie open source è disponibile una grande quantità di documentazione, forum e discussioni derivanti dall'ecosistema e risorse pubbliche. Per beneficiare appieno dell'approccio open source, è necessario stabilire una roadmap delle risorse attuali e degli obiettivi desiderati per impostare un programma coerente di competenze, metodi e strumenti di sviluppo all'interno dei team.

Settori di applicazione È necessario stabilire le aree in cui il programma sarà applicato e come migliorerà la qualità e l'efficienza del codice e delle pratiche. Ad esempio, il programma non avrà gli stessi benefici se c'è un solo sviluppatore che lavora su componenti open source, o se l'intero ciclo di vita dello sviluppo è ottimizzato per includere le migliori pratiche open source.

È necessario definire l'ambito di applicazione dello sviluppo open source: componenti tecnici, applicazioni, modernizzazione o creazione di nuovi sviluppi. Esempi di pratiche di sviluppo che possono beneficiare dell'open source sono:

- Amministrazione del cloud.
- Applicazioni cloud-native, come innovare con queste tecnologie.
- DevOps, Continuous Integration / Continuous Delivery.

Categorie

- Competenze e risorse necessarie per sviluppare software open source: proprietà intellettuale, licenze, pratiche.
- Competenze e risorse necessarie per sviluppare software utilizzando componenti, linguaggi e tecnologie open source.
- Competenze e risorse necessarie per utilizzare metodi e processi open source.

Valutazione delle opportunità

Gli strumenti open source sono sempre più diffusi tra gli sviluppatori. Questa Activity affronta la necessità di evitare la proliferazione di strumenti eterogenei all'interno di un team di sviluppo. Aiuta a definire una politica in questo ambito. Aiuta a ottimizzare la formazione e la creazione di esperienza. Un inventario delle competenze viene utilizzato per l'assunzione, la formazione

e la pianificazione della successione nel caso in cui un dipendente che ricopre un ruolo chiave lasci l'azienda.

Avremmo bisogno di una metodologia per mappare le competenze di sviluppo del software open source.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- È presente una descrizione della catena di produzione dell'open source (la "filiera del software"),
- Esiste un piano (o una lista di desideri) per la razionalizzazione delle risorse di sviluppo,
- Esiste un inventario delle competenze che riassume le competenze, l'istruzione e l'esperienza degli attuali sviluppatori,
- Esiste un elenco dei desideri di formazione e un programma che si occupa delle carenze di competenze,
- Esiste un elenco di best practice di sviluppo open source mancanti e un piano per applicarle.

Raccomandazioni

- Iniziare in modo semplice, far crescere l'analisi e la roadmap in modo costante.
- Nel reclutamento, ponete una forte enfasi sulle competenze e sull'esperienza open source. È sempre più facile quando le persone hanno già un DNA open source piuttosto che formare e addestrare le persone.
- Verificate i programmi di formazione dei fornitori di software e delle scuole open source.

Risorse

Ulteriori informazioni:

- Un'introduzione a [cos'è uno Skills Inventory?](#) di Robert Tanner.
- Un articolo sulle competenze open source: [5 competenze open source per migliorare il vostro gioco e il vostro curriculum](#)

Questa attività può includere risorse e competenze tecniche quali:

- **Linguaggi popolari** (come Java, PHP, Perl, Python).
- **Framework open source** (Spring, AngularJS, Symfony) e strumenti di test.
- **Metodi di sviluppo e best practice** Agile, DevOps e open source.

Attività associate:

- [GGI-A-28 Prospettiva HR](#)

5 Attività dell'obiettivo fiducia (Trust Goal)

5.1 Gestire la conformità legale

Activity ID: [GGI-A-21](#).

Descrizione

Le organizzazioni devono implementare un processo di conformità legale per tutelare il loro utilizzo e la loro partecipazione a progetti open source.

La gestione matura e professionale della conformità legale, all'interno dell'organizzazione e lungo la catena di fornitura, riguarda:

- Esecuzione di un'analisi approfondita della proprietà intellettuale che includa l'identificazione della licenza e la verifica della compatibilità.
- Garantire che l'organizzazione possa utilizzare, integrare, modificare e ridistribuire in modo sicuro componenti open source come parte dei suoi prodotti o servizi.
- Fornire a dipendenti e collaboratori un processo trasparente su come creare e contribuire al software open source.

Analisi della composizione del software (SCA): Una parte significativa dei problemi legali e di proprietà intellettuale deriva dall'uso di componenti rilasciati sotto licenze che sono incompatibili tra loro o incompatibili con il modo in cui l'organizzazione vuole usare e ridistribuire i componenti. La SCA è il primo passo per risolvere questi problemi, perché "bisogna conoscere il problema per poterlo risolvere". Il processo consiste nell'identificare tutti i componenti coinvolti in un progetto in un documento di distinta base, comprese le dipendenze di build e test.

Controllo delle licenze: Un processo di controllo delle licenze utilizza uno strumento per analizzare automaticamente la base di codice e identificare le licenze e i diritti d'autore al suo interno. Se eseguito regolarmente e idealmente integrato nelle catene di compilazione e integrazione continue, consente di individuare tempestivamente i problemi di proprietà intellettuale.

Valutazione delle opportunità

Con l'uso sempre più diffuso degli OSS nei sistemi informativi di un'organizzazione, è essenziale valutare e gestire la potenziale esposizione legale.

Tuttavia, la verifica delle licenze e dei diritti d'autore può essere complicata e costosa. Gli sviluppatori devono essere in grado di verificare rapidamente la proprietà intellettuale e le questioni legali. Avere un team e un responsabile aziendale dedicato alla proprietà intellettuale e alle questioni legali garantisce una gestione proattiva e coerente delle questioni legali, aiuta a garantire l'uso e i contributi dei componenti open source e fornisce una chiara visione strategica.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Per i progetti è disponibile un semplice processo di verifica delle licenze.
- Per i progetti è disponibile un semplice processo di verifica della proprietà intellettuale.
- All'interno dell'organizzazione esiste un team o una persona responsabile della conformità legale.
- Sono previsti audit periodici per valutare la conformità legale.

Altri modi per impostare i punti di verifica:

- Esiste un semplice processo di verifica delle licenze.
- Esiste un team legale/IP che può essere consultato come nell'attività #13.
- Tutti i progetti forniscono le informazioni richieste affinché le persone possano utilizzarle e contribuire al progetto.
- C'è un contatto nel team per le domande relative alla proprietà intellettuale e alle licenze.
- Esiste un dipendente aziendale dedicato alla proprietà intellettuale e alle licenze.
- Esiste un team dedicato alle domande relative alla proprietà intellettuale ed alle licenze.

Strumenti

- [ScanCode](#)
- [Fossology](#)
- [SW360](#)
- [Fossa](#)
- [OSS Review Toolkit](#)

Raccomandazioni

- Informare le persone sui rischi associati alla concessione di licenze in conflitto con gli obiettivi aziendali.
- Proporre una soluzione semplice per i progetti per impostare il controllo delle licenze sul loro codice.
- Comunicare la sua importanza e aiutare i progetti ad aggiungerlo ai loro sistemi di CI.
- Fornire un modello o linee guida ufficiali per la struttura del progetto.
- Impostare controlli automatici per assicurarsi che tutti i progetti siano conformi alle linee guida.
- Considerare la possibilità di condurre un audit interno per identificare le licenze dell'infrastruttura aziendale.
- Fornire una formazione di base sulla proprietà intellettuale e sulle licenze ad almeno una persona per team.
- Fornire una formazione completa sulla proprietà intellettuale e sulle licenze per il responsabile.
- Creare un processo di escalation delle questioni relative alla proprietà intellettuale e alle licenze per il responsabile.

Ricordate che la conformità non riguarda solo la legge, ma anche la proprietà intellettuale. Ecco quindi alcune domande che possono aiutare a comprendere le conseguenze della conformità legale:

- Se distribuisco un componente open source e non rispetto le condizioni della licenza, violo la licenza stessa --> implicazioni legali.
- Se utilizzo un componente open source all'interno di un progetto che desidero distribuire/pubblicare, tale licenza può obbligare a rendere visibili elementi del codice che non voglio rendere open source --> Impatto sulla riservatezza per il vantaggio tattico della mia azienda e con terze parti (implicazioni legali).
- È una discussione aperta il fatto che l'uso di una licenza open source per un progetto che voglio pubblicare garantisca una proprietà intellettuale rilevante --> implicazioni sulla proprietà intellettuale.
- Se rendo un progetto open source *prima di* qualsiasi processo di brevettazione, questo *probabilmente* esclude la creazione di brevetti riguardanti il progetto --> implicazioni sulla proprietà intellettuale.
- Se rendo un progetto open source *dopo* qualsiasi processo di brevettazione, questo *probabilmente* permette la creazione di brevetti (difensivi) relativi a quel progetto --> potenziale proprietà intellettuale.
- In progetti complessi che coinvolgono molti componenti con molte dipendenze, la moltitudine di licenze open source può presentare incompatibilità tra le licenze --> implicazioni legali (cfr. Issue #23).

Risorse

- Un ampio elenco di strumenti è disponibile nella pagina [Existing OSS compliance group](#).
- [Recommended Open Source Compliance Practices for the enterprise](#). Un libro di Ibrahim Haddad, della Linux Foundation, sulle pratiche di conformità all'open source per le aziende. [OpenChain Project](#)

5.2 Gestire le vulnerabilità del software

Activity ID: [GGI-A-22](#).

Descrizione

Il codice di uno sviluppatore è sicuro quanto la parte meno sicura del suo codice. Casi recenti (ad esempio [heartbleed](#) ¹, [equifax](#) ²) hanno dimostrato l'importanza di verificare la presenza di vulnerabilità in parti di codice non direttamente sviluppate in azienda. Le conseguenze delle esposizioni vanno dalle fughe di dati (con un notevole impatto sulla reputazione) agli attacchi di malware (ransomware) e all'indisponibilità di servizi a rischio per l'azienda.

Il software open source è noto per avere una migliore gestione delle vulnerabilità rispetto al software proprietario, soprattutto perché:

- Sempre maggiore attenzione è rivolta a trovare e risolvere i problemi del codice e dei processi aperti.
- I progetti open source risolvono le vulnerabilità e rilasciano patch e nuove versioni molto più velocemente.

Ad esempio, uno [studio di WhiteSource](#) sul software proprietario ha mostrato che il 95% delle vulnerabilità riscontrate nei loro componenti open source aveva già rilasciato una correzione al momento dell'analisi. Il problema, quindi, è **gestire meglio le vulnerabilità sia nella base di codice che nelle sue dipendenze**, indipendentemente dal fatto che siano closed o open source.

Per ridurre questi rischi, è necessario istituire un programma di valutazione delle risorse software e un processo di verifica delle vulnerabilità eseguito regolarmente. Implementare strumenti che avvisino i team interessati, gestiscano le vulnerabilità note e prevengano le minacce derivanti dalle dipendenze del software.

Valutazione delle opportunità

Qualsiasi azienda che utilizza software deve tenere sotto controllo le sue vulnerabilità:

- la sua infrastruttura (ad es. infrastruttura Cloud, infrastruttura di rete, archivi di dati),
- le sue applicazioni aziendali (HR, strumenti CRM, gestione dei dati interni e dei clienti),
- il suo codice interno: ad esempio il sito web dell'azienda, i progetti di sviluppo interno, ecc,
- e tutte le dipendenze dirette e indirette di software e servizi.

Il ROI delle vulnerabilità è poco conosciuto finché non accade qualcosa di grave. Per stimare il costo reale delle vulnerabilità bisogna considerare le conseguenze di una grave violazione dei dati o dell'indisponibilità dei servizi.

Allo stesso modo, è necessario evitare a tutti i costi una cultura di segretezza e occultamento dei problemi di sicurezza all'interno dell'azienda. Al contrario, le informazioni sullo stato di vulnerabilità devono essere condivise e discusse per trovare le risposte migliori dalle persone giuste, dagli sviluppatori ai dirigenti di livello superiore.

I vantaggi della prevenzione degli attacchi informatici attraverso un'attenta gestione delle vulnerabilità del software sono molteplici:

- Evitare impatti negativi sulla reputazione,
- Evitare perdite dovute ad azioni di Exploiting (DDoS, Ransomware, tempo per ricostruire un sistema IT alternativo dopo un attacco),
- Rispettare le norme sulla protezione dei dati.

La gestione delle vulnerabilità del software OSS è solo una parte del più ampio processo di cybersecurity che affronta globalmente la sicurezza dei sistemi e dei servizi dell'organizzazione.

Valutazione dei progressi

Dovrebbe esserci una persona o un team dedicato al monitoraggio delle vulnerabilità e all'implementazione di processi facili da usare su cui gli sviluppatori possano fare affidamento. La valutazione delle vulnerabilità è una parte standard del processo di integrazione continua e le persone sono in grado di monitorare lo stato attuale del rischio in una dashboard dedicata.

¹<https://it.wikipedia.org/wiki/Heartbleed>

²<https://arstechnica.com/information-technology/2017/09/massive-equifax-breach-caused-by-failure-to-patch-two-month-old-bug/>

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- L'attività è coperta quando tutti i software e i servizi interni sono valutati e monitorati per le vulnerabilità note.
- L'attività è coperta quando nella catena di produzione del software vengono implementati uno strumento e un processo dedicati per prevenire l'introduzione di problemi nelle routine di sviluppo quotidiane.
- Una persona o un team è responsabile della valutazione del rischio di CVE/vulnerabilità rispetto all'esposizione.
- Una persona o un team è responsabile dell'invio di CVE/vulnerability alle persone interessate (SysOps, DevOps, sviluppatori, ecc.).

Strumenti

- Strumenti GitHub
 - GitHub fornisce linee guida e strumenti per proteggere il codice ospitato sulla piattaforma. Per ulteriori informazioni, consultare [GitHub docs](#).
 - GitHub fornisce [Dependabot](#) per identificare automaticamente le vulnerabilità nelle dipendenze.
- [Eclipse Steady](#) è uno strumento gratuito e open source che analizza i progetti Java e Python alla ricerca di vulnerabilità e aiuta gli sviluppatori a mitigarle.
- [OWASP dependency-check](#): uno scanner di vulnerabilità open source.
- [OSS Review Toolkit](#): un orchestratore open source in grado di raccogliere avvisi di sicurezza per le dipendenze utilizzate dai servizi di dati sulle vulnerabilità configurati.

Risorse

- Il [database delle vulnerabilità del MITRE](#) dei CVE. Si veda anche il [database della sicurezza del NIST](#) delle NVD e risorse satellite come [CVE Details](#).
- Si veda anche questa nuova iniziativa di Google: le [Vulnerabilità open source](#).
- Il gruppo di lavoro OWASP pubblica un elenco di scanner di vulnerabilità [sul proprio sito web](#), sia del mondo commerciale che di quello open source.
- J. Williams e A. Dabirsiaghi. The unfortunate reality of insecure libraries, 2012.
- [Rilevazione, valutazione e mitigazione delle vulnerabilità nelle dipendenze open source](#), Serena Elisa Ponta, Henrik Plate & Antonino Sabetta, Empirical Software Engineering volume 25, pagine 3175-3215(2020).
- [A Manually-Curated Dataset of Fixes to Vulnerabilities of open source Software](#), Serena E. Ponta, Henrik Plate, Antonino Sabetta, Michele Bezzi, Cédric Dangremont. Esiste anche un [toolkit in sviluppo per implementare il suddetto dataset](#).

5.3 Gestire le dipendenze del software

Activity ID: [GGI-A-23](#).

Descrizione

Un programma di *identificazione delle dipendenze* cerca le dipendenze effettivamente utilizzate all'interno del codice base. Di conseguenza, l'organizzazione deve stabilire e mantenere un elenco di dipendenze note e osservare l'evoluzione dei fornitori identificati.

Stabilire e mantenere un elenco di dipendenze conosciute è un fattore abilitante e un prerequisito per:

- Controllo della proprietà intellettuale e delle licenze: alcune licenze non possono essere mescolate, nemmeno come dipendenza. È necessario conoscere le dipendenze per valutare i rischi legali associati.
- Gestione delle vulnerabilità: l'intero software è debole quanto la sua parte più debole: si veda l'esempio della falla [Heartbleed flaw](#). È necessario conoscere le sue dipendenze per valutare i rischi di sicurezza associati.
- Ciclo di vita e sostenibilità: una comunità attiva sul progetto di dipendenza è un segnale positivo per la correzione di bug, ottimizzazioni e nuove funzionalità.

- Selezione ponderata delle dipendenze utilizzate, in base a criteri di "maturità": l'obiettivo è utilizzare componenti open source sicuri, con una base di codice sana e ben mantenuta, una comunità viva, attiva e reattiva che accetti contributi esterni, ecc.

Valutazione delle opportunità

L'identificazione e la tracciabilità delle dipendenze è una fase necessaria per ridurre i rischi associati al riutilizzo del codice. Inoltre, l'implementazione di strumenti e processi per gestire le dipendenze del software è un prerequisito per gestire correttamente la qualità, la conformità e la sicurezza.

Considerate le seguenti domande:

- Qual è il rischio dell'azienda (costo, reputazione, ecc.) se il software viene danneggiato, attaccato o citato in giudizio?
- La codebase è considerata critica per le persone, l'organizzazione o l'azienda?
- Cosa succede se un componente da cui dipende un'applicazione cambia il suo repository?

Il primo passo consiste nell'implementare uno strumento di analisi della composizione del software (software composition analysis, SCA). Per una SCA completa o una mappatura delle dipendenze può essere necessario il supporto di società di consulenza specializzate.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Le dipendenze sono identificate in tutto il codice sviluppato internamente.
- Le dipendenze vengono identificate in tutto il codice esterno eseguito all'interno dell'azienda.
- È disponibile un'analisi della composizione del software o una procedura di identificazione delle dipendenze facile da impostare, che i progetti possono aggiungere al loro processo di integrazione continua.
- Vengono utilizzati strumenti di analisi delle dipendenze.

Strumenti

- [OWASP Dependency check](#): dependency-Check è uno strumento di analisi della composizione del software (SCA) che tenta di rilevare le vulnerabilità pubblicamente divulgate contenute nelle dipendenze di un progetto.
- [OSS Review Toolkit](#): una suite di strumenti per la revisione delle dipendenze del software open source.
- [Fossa](#): analisi delle dipendenze veloce, portatile e affidabile. Supporta la scansione delle licenze e delle vulnerabilità. È indipendente dal linguaggio; si integra con oltre 20 sistemi di compilazione.
- [Software 360](#).
- [Eclipse Dash license tool](#): prende un elenco di dipendenze e chiede a [ClearlyDefined](#) di controllare le loro licenze.
- [Il progetto FOSSology](#): FOSSology è un progetto open source con la missione di promuovere la conformità alle licenze open source.

Raccomandazioni

- Condurre audit regolari sulle dipendenze e sui requisiti di proprietà intellettuale per ridurre i rischi legali.
- L'ideale è integrare la gestione delle dipendenze nel processo di integrazione continua, in modo che i problemi (nuove dipendenze, incompatibilità di licenze) vengano identificati e risolti il prima possibile.
- Tenere traccia delle vulnerabilità legate alle dipendenze, tenere informati utenti e sviluppatori.
- Informare le persone sui rischi associati a licenze errate.
- Proporre una soluzione semplice per i progetti per impostare il controllo delle licenze sul loro codice.

- Comunicare la sua importanza e aiutare i progetti ad aggiungerlo ai loro sistemi di CI.
- Impostare un KPI visibile per i rischi legati alle dipendenze.

Risorse

- Pagina del gruppo [Strumenti di conformità alle licenze OSS](#) esistente.
- [\[Free and Open Source Software licence Compliance: Tools for Software Composition Analysis\]](#), di Philippe Ombredanne, nexB Inc.
- [Software Sustainability Maturity Model](#).
- [CHAOS](#): Community Health Analytics Open Source Software.

5.4 Gestire gli indicatori chiave

Activity ID: [GGI-A-24](#).

Descrizione

Questa attività raccoglie e monitora una serie di indicatori che supportano le decisioni manageriali quotidiane e le opzioni strategiche relative al software open source gestito professionalmente.

Le metriche chiave relative al software open source costituiscono lo sfondo del buon andamento dei programmi di governance. L'attività prevede la selezione di alcuni indicatori, la loro pubblicazione ai team e alla direzione e l'invio di aggiornamenti regolari sull'iniziativa, ad esempio tramite una newsletter o un notiziario aziendale.

Questa attività richiede:

- presenza degli stakeholder per discutere e definire gli obiettivi del programma,
- l'implementazione di uno strumento di misurazione e raccolta dati connesso all'infrastruttura di sviluppo,
- la pubblicazione di almeno una dashboard per gli stakeholder e per tutte le persone coinvolte nell'iniziativa.

Gli indicatori si basano su dati che devono essere raccolti da fonti pertinenti. Fortunatamente, ci sono molte fonti per l'ingegneria del software open source. Alcuni esempi sono:

- l'ambiente di sviluppo, la catena di produzione CI/CD,
- il dipartimento Risorse Umane,
- gli strumenti di test e di analisi della composizione del software,
- i repository.

Esempi di indicatori sono:

- Numero di dipendenze risolte, visualizzate per tipo di licenza.
- Numero di dipendenze obsolete/vulnerabili.
- Numero di problemi di licenza/ip rilevati.
- Contributi a progetti esterni.
- Bug open time.
- Numero di collaboratori di un componente, numero di commit, ecc.

Questa attività consiste nel definire i requisiti e le esigenze di misurazione e nell'implementare un cruscotto che mostri in modo semplice ed efficace i principali indicatori del programma.

Valutazione delle opportunità

Gli indicatori chiave aiutano a capire e a gestire meglio le risorse dedicate al software open source e a misurare i risultati per comunicare in modo efficace e raccogliere tutti i benefici dell'investimento. Grazie a un'ampia comunicazione, un maggior numero di persone può seguire l'iniziativa e sentirsi coinvolto, il che, in ultima analisi, la rende un obiettivo e una priorità dell'organizzazione.

Sebbene ogni attività abbia dei criteri di valutazione che aiutano a rispondere alle domande sui progressi raggiunti, c'è ancora bisogno di un monitoraggio fatto di numeri e indicatori quantitativi.

Che si tratti di una piccola startup o di una grande azienda globale, le metriche chiave aiutano a mantenere i team concentrati e a monitorare le prestazioni. Le metriche sono fondamentali perché supportano il processo decisionale e sono la base per il monitoraggio delle decisioni già prese.

Con numeri e grafici semplici e pratici, i membri di tutta l'organizzazione saranno in grado di seguire e sincronizzare gli sforzi relativi all'open source, rendendolo una preoccupazione e un'azione condivisa. Questo permette anche ai vari attori di entrare meglio nel corso, di contribuire al progetto e di ottenere i benefici complessivi.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- È stato stabilito un elenco di metriche e di modalità di raccolta.
- Vengono utilizzati strumenti per raccogliere, memorizzare, elaborare e visualizzare gli indicatori.
- Tutti i partecipanti hanno a disposizione una dashboard generale che mostra i progressi dell'iniziativa.

Strumenti

- [GrimoireLab](#) di Bitergia.
- Anche gli strumenti di BI generici (elasticsearch, grafana, visualizzazioni R/Python...) sono adatti, se i connettori appropriati sono impostati in base agli obiettivi definiti.

Raccomandazioni

- Scrivere gli obiettivi e la roadmap della governance open source.
- Comunicare all'interno dell'azienda le azioni e lo stato dell'iniziativa.
- Coinvolgere le persone nella definizione dei KPI, per assicurarsi che
 - siano ben compresi,
 - forniscano una visione completa delle esigenze e delle
 - che vengano prese in considerazione e seguite.
- Costruire almeno una dashboard che possa essere visualizzata da tutti (ad esempio su uno schermo nella stanza), con indicatori essenziali per mostrare i progressi e la situazione generale.

Risorse

- La [comunità CHAOSS](#) ha molti buoni riferimenti e risorse relative agli indicatori open source.
- Verificate le metriche per [Attributi del progetto](#) dalla [metodologia OW2 Market Readiness Levels](#).
- [A New Way of Measuring Openness: The Open Governance Index](#) di Liz Laffan è una lettura interessante sull'apertura nei progetti open source.
- [Governance Indicators: A Users' Guide](#) è la guida delle Nazioni Unite sugli indicatori di governance. Sebbene sia applicata alla democrazia, alla corruzione e alla trasparenza delle nazioni, vale la pena di leggere le basi della misurazione e degli indicatori applicati alla governance.

5.5 Eseguire revisioni del codice

Activity ID: [GGI-A-44](#).

Descrizione

La revisione del codice è un'attività di routine che prevede la revisione manuale e/o automatizzata del codice sorgente di un'applicazione prima di rilasciare un prodotto o consegnare un progetto al cliente. Nel caso del software open-source, la revisione del codice

non si limita a individuare opportunamente gli errori, ma è un approccio integrato allo sviluppo collaborativo condotto a livello di team.

La revisione del codice dovrebbe riguardare sia il codice sviluppato internamente sia quello riutilizzato da fonti esterne, in quanto migliora la fiducia generale nel codice e ne rafforza la proprietà. È anche un modo eccellente per migliorare le competenze e le conoscenze globali all'interno del team e per promuovere la collaborazione tra i membri.

Valutazione delle opportunità

Le revisioni del codice sono preziose ogni volta che l'organizzazione sviluppa software o riutilizza software esterni. Pur essendo una fase standard del processo di ingegneria del software, le revisioni del codice nel contesto dell'open-source apportano vantaggi specifici quali:

- Quando si pubblica il codice sorgente interno, permette di verificare che vengano rispettate le linee guida sulla qualità.
- Quando si contribuisce a un progetto open source esistente, permette di verificare che le linee guida del progetto in questione siano rispettate.
- La documentazione pubblica viene aggiornata di conseguenza.

È anche un'ottima occasione per condividere e far rispettare alcune regole di conformità legale della vostra azienda, come ad esempio:

- Non rimuovete mai le intestazioni di licenza o i diritti d'autore esistenti nel codice open source riutilizzato.
- Non copiare e incollare il codice sorgente da Stack Overflow senza previa autorizzazione del team legale.
- Includere la linea di copyright corretta quando richiesto.

Le revisioni del codice garantiscono la fiducia e la sicurezza del codice. Se le persone non sono sicure della qualità o dei rischi potenziali dell'uso di un prodotto software, dovrebbero effettuare revisioni del codice e raccogliere feedback dai colleghi.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- La revisione del codice open source è riconosciuta come un passo necessario.
- Sono previste revisioni del codice open source (regolarmente o in momenti critici).
- È stato definito e accettato collettivamente un processo di revisione del codice open-source.
- Le revisioni del codice open-source sono una parte standard del processo di sviluppo.

Raccomandazioni

- La revisione del codice è un compito collettivo che funziona meglio in un buon ambiente collaborativo.
- Non esitate a utilizzare gli strumenti e i modelli esistenti nel mondo open-source, dove la revisione del codice è uno standard da anni (decenni).

Risorse

- [What is Code Review?](#): una lettura didattica sulla revisione del codice trovata sulla Open Practice Library di Red Hat.
- [Best Practices for Code Reviews](#): un'altra prospettiva interessante su cosa sia la revisione del codice.

6 Attività dell'obiettivo culturale (Culture Goal)

6.1 Promuovere le migliori pratiche di sviluppo open source

Activity ID: [GGI-A-25](#).

Descrizione

Questa attività consiste nel definire, promuovere attivamente e implementare le migliori pratiche open source all'interno dei team di sviluppo.

Come punto di partenza, si potrebbero prendere in considerazione i seguenti argomenti:

- Documentazione per utenti e sviluppatori.
- Organizzazione corretta del progetto su un repository accessibile al pubblico.
- Promuovere e implementare il riutilizzo controllato.
- Fornire una documentazione completa e aggiornata sul prodotto.
- Configuration Management: flussi di lavoro git, modelli collaborativi.
- Release management: rilascio anticipato e frequente, versioni stabili o di sviluppo, ecc.

I progetti OSS hanno un modus operandi speciale, simile a un [bazar](#). Per consentire e favorire questa collaborazione e questa mentalità, si raccomandano alcune pratiche che facilitano lo sviluppo collaborativo e decentralizzato e i contributi di sviluppatori terzi...

Documenti per community Assicuratevi che tutti i progetti all'interno dell'azienda propongano i seguenti documenti:

- README -- descrizione rapida del progetto, come interagire, link alle risorse.
- Contributing -- introduzione per le persone disposte a contribuire.
- Code Of Conduct: cosa è accettabile o meno come comportamento all'interno della comunità.
- LICENSE -- la licenza predefinita del repository.

REUSE best practices [REUSE](#) è un'iniziativa della [Free Software Foundation Europe](#) per migliorare il riutilizzo del software e semplificare la conformità agli OSS e alle licenze.

Valutazione delle opportunità

Sebbene dipenda fortemente dalla conoscenza comune dell'OSS da parte del team, la formazione delle persone e la creazione di processi che impongano queste pratiche è sempre utile. È ancora più importante quando:

- potenziali utenti e contributori non sono noti,
- gli sviluppatori non sono abituati allo sviluppo open source.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Il progetto stabilisce un elenco di best practice open source da rispettare.
- Il progetto monitora il suo allineamento con le migliori pratiche.
- Il team di sviluppo si è sensibilizzato al rispetto delle best practice OSS.
- Le nuove best practice vengono valutate regolarmente e si cerca di implementarle.

Strumenti

- Lo strumento [REUSE helper tool](#) aiuta a rendere un repository conforme alle migliori pratiche [REUSE](#). Può essere incluso in molti processi di sviluppo per confermare lo stato attuale.
- [ScanCode](#) ha la capacità di elencare tutti i documenti comunitari e legali presenti nell'archivio: vedere [descrizione della funzione](#).
- GitHub ha una bella funzione per [controllare i documenti mancanti della comunità](#). Si trova nella pagina del repository > "Insights" > "Community".

Raccomandazioni

- L'elenco delle migliori pratiche dipende dal contesto e dall'ambito del programma e deve essere rivalutato regolarmente in un'ottica di miglioramento continuo. Le pratiche devono essere monitorate e valutate regolarmente per monitorare i progressi.
- Formare le persone sul riutilizzo degli OSS (come consumatori) e sugli ecosistemi (come contributori).
- Implementare REUSE.software come nell'activity #14.
- Impostare un processo per gestire i rischi legali associati al riutilizzo e ai contributi.
- Incoraggiare esplicitamente le persone a contribuire a progetti esterni.
- Fornire un modello o linee guida ufficiali per la struttura del progetto.
- Impostare controlli automatici per assicurarsi che tutti i progetti siano conformi alle linee guida.

Risorse

- [elenco di best practice open source di OW2](#) dalla metodologia di valutazione Market Readiness Levels.
- [Sito ufficiale di REUSE](#) con specifiche, tutorial e FAQ.
- [Linee guida della comunità di GitHub](#).
- Un esempio di [migliori pratiche di gestione della configurazione con GitHub](#).

6.2 Contribuire a progetti open source

Activity ID: [GGI-A-26](#).

Descrizione

Contribuire a progetti open source di libero utilizzo è uno dei principi chiave della buona governance. Il punto è evitare di essere un semplice consumatore passivo e dare un contributo ai progetti. Quando le persone aggiungono una funzionalità o correggono un bug per i propri scopi, dovrebbero renderlo abbastanza generico da contribuire al progetto. Agli sviluppatori deve essere concesso il tempo necessario per contribuire.

Questa attività copre i seguenti ambiti:

- Lavorare con progetti open source upstream.
- Segnalazione di bug e richieste di funzionalità.
- Contribuire al codice e alla correzione di bug.
- Partecipare alle mailing list della community.
- Condividere le esperienze.

Valutazione delle opportunità

I principali benefici di questa attività sono:

- Aumenta la conoscenza generale e l'impegno verso l'open source all'interno dell'azienda, in quanto le persone iniziano a contribuire e a essere coinvolte in progetti open source. Si percepisce un senso di utilità pubblica e si migliora la propria reputazione personale.
- L'azienda aumenta la propria visibilità e reputazione man mano che i contributi si fanno strada nel progetto. Questo dimostra che l'azienda è effettivamente coinvolta nell'open source, contribuisce e promuove correttezza e trasparenza.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste un percorso chiaro e ufficiale per le persone disposte a contribuire.
- Gli sviluppatori sono incoraggiati a contribuire ai progetti open source che utilizzano.
- È in atto un processo per garantire la conformità legale e la sicurezza dei contributi da parte degli sviluppatori.

- KPI: Volume dei contributi esterni (codice, mailing list, issue...) da parte di individui, team o entità.

Strumenti

Può essere utile seguire i contributi, sia per tenere traccia di ciò che viene apportato sia per poter comunicare l'impegno dell'azienda. A questo scopo si possono utilizzare dashboard e software di tracciamento delle attività. Controllare:

- [GrimoireLab](#) di Bitergia
- [ScanCode](#)

Raccomandazioni

Incoraggiare le persone all'interno dell'entità a contribuire a progetti esterni:

- Permettere di scrivere correzioni di bug e funzionalità generiche e ben testate e di contribuire alla community.
- Fornire formazione alle persone sul contributo alle comunità open source. Questo riguarda sia le competenze tecniche (migliorare le conoscenze del team) che la community (appartenenza alle community open source, codice di condotta, ecc.).
- Fornire formazione su questioni legali, di proprietà intellettuale e tecniche e creare un contatto all'interno dell'azienda che possa aiutarvi su questi temi in caso di dubbi.
- Fornire incentivi per i lavori pubblicati.
- Si noti che i contributi dell'azienda/ente rifletteranno la qualità del codice e il suo coinvolgimento, quindi assicuratevi che il vostro team di sviluppo fornisca codice sufficientemente buono.

Risorse

- L'iniziativa [CHAOSS](#) della Linux Foundation offre alcuni strumenti e indicazioni su come tracciare i contributi allo sviluppo.

6.3 Appartenere alla comunità open source

Activity ID: [GGI-A-27](#).

Descrizione

Questa attività mira a sviluppare tra gli sviluppatori un senso di appartenenza alla comunità open source. Come in ogni community, le persone e le entità devono partecipare e contribuire all'insieme. Rafforza i legami tra i professionisti e porta sostenibilità e attività all'ecosistema. Da un punto di vista più tecnico, permette di scegliere le priorità e la roadmap dei progetti, migliora il livello di conoscenza generale e di consapevolezza tecnica.

Questa attività copre quanto segue:

- **Identificare gli eventi** a cui vale la pena partecipare. Mettere in contatto le persone, conoscere le nuove tecnologie e creare una rete di contatti sono fattori chiave per ottenere tutti i vantaggi dell'open-source.
- Considerate **l'adesione a fondazioni**. Le fondazioni e le organizzazioni open-source sono una componente chiave dell'ecosistema open-source. Forniscono risorse tecniche e organizzative ai progetti e sono un buon luogo neutrale per gli sponsor per discutere problemi e soluzioni comuni o per lavorare sugli standard.
- Osservate i **gruppi di lavoro**. I gruppi di lavoro sono spazi di lavoro collaborativi neutrali in cui gli esperti interagiscono su un dominio specifico come l'IoT, la modellazione o la scienza. Sono un meccanismo molto efficiente ed economico per affrontare insieme problemi comuni, anche se specifici del dominio.
- **Partecipazione al bilancio**. In definitiva, il denaro è il catalizzatore. Pianificate le spese necessarie, date alle persone un tempo retribuito per queste attività, anticipate le azioni successive, in modo che il programma non debba fermarsi dopo pochi mesi per mancanza di fondi.

Valutazione delle opportunità

L'open source funziona meglio quando è in relazione con la comunità open source in generale. Facilita la correzione di bug, la condivisione di soluzioni, ecc.

È anche un buon modo per le aziende di dimostrare il loro sostegno ai valori dell'open-source. Comunicare il coinvolgimento dell'azienda è importante sia per la sua reputazione che per l'ecosistema open-source.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- È stato stilato un elenco di eventi a cui le persone potrebbero partecipare.
- È disponibile un monitoraggio delle conferenze pubbliche tenute dai membri del team.
- Le persone possono inviare richieste di partecipazione agli eventi.
- Le persone possono presentare progetti da sponsorizzare.

Raccomandazioni

- Fate un sondaggio tra le persone per sapere quali sono gli eventi che preferiscono o che sarebbero più utili per il loro lavoro.
- Creare comunicazioni interne (newsletter, centro risorse, inviti...) in modo che le persone sappiano delle iniziative e possano partecipare.
- Assicuratevi che queste iniziative possano andare a beneficio di vari tipi di persone (sviluppatori, amministratori, supporto...), non solo dei dirigenti di livello dirigenziale (C-Level).

Risorse

- [Cosa spinge uno sviluppatore a contribuire al software open source?](#) Un articolo di Michael Sweeney su clearcode.cc.
- [Perché le aziende contribuiscono all'open source](#) Un articolo di Velichka Atanasova di VMware.
- [Perché i vostri dipendenti dovrebbero contribuire all'open source](#) Una buona lettura di Robert Kowalski di CloudBees.
- [7 modi in cui la vostra azienda può sostenere l'open source](#) Un articolo di Simon Phipps per InfoWorld.
- [Eventi: la forza vitale dell'open source](#) Un articolo di Donna Benjamin di RedHat.

6.4 Prospettiva delle risorse umane

Activity ID: [GGI-A-28](#).

Descrizione

Il passaggio alla cultura open source ha un impatto profondo sulle risorse umane:

- **Nuovi processi e contratti:** I contratti devono essere adattati per consentire e promuovere i contributi esterni. Ciò include le questioni relative alla proprietà intellettuale e alle licenze per il lavoro svolto all'interno dell'azienda, ma anche la possibilità per il dipendente o l'appaltatore di avere i propri progetti.
- **Diversi tipi di persone:** Le persone che lavorano con l'open source hanno spesso incentivi e mentalità diverse rispetto a quelle delle aziende proprietarie. I processi e le mentalità devono adattarsi a questo paradigma orientato alla reputazione della comunità, per attrarre nuovi tipi di talenti e trattenerli.
- **Sviluppo della carriera:** è necessario offrire un percorso di carriera che valorizzi i dipendenti per le loro competenze tecniche e trasversali e per le competenze attese dall'organizzazione (collaborazione per guidare gli sforzi della comunità, comunicazione per agire come portavoce dell'azienda, ecc.) In ogni caso, le risorse umane hanno un ruolo chiave nel rendere l'open source un obiettivo culturale.

Forza lavoro Per uno sviluppatore che ha lavorato a lungo sulla stessa soluzione proprietaria, passare all'open source può sembrare un cambiamento e richiedere un adattamento. Ma per la maggior parte degli sviluppatori il software open source porta solo vantaggi.

Gli sviluppatori che escono dalla scuola o dall'università oggi hanno sempre lavorato sull'open source. All'interno di un'azienda, la grande maggioranza degli sviluppatori utilizza linguaggi open source e importa librerie o snippet open source ogni giorno. È infatti molto più facile incollare righe di codice open source in un programma che attivare il processo di sourcing interno, che passa attraverso molteplici convalide attraverso la linea manageriale.

L'open source rende il lavoro dello sviluppatore più interessante, perché lo sviluppatore è sempre alla ricerca di ciò che i suoi colleghi al di fuori dell'azienda hanno inventato, e quindi rimane all'avanguardia della tecnologia.

Per un'organizzazione, è necessaria una strategia delle risorse umane per 1/ qualificare o riqualificare la forza lavoro esistente 2/ riflettere e posizionare l'azienda sull'assunzione di nuovi talenti, e dunque comunicare qual è l'attrattiva dell'azienda quando si tratta di open source.

Trovare persone con una buona mentalità FLOSS, che capiscano già il codice e sappiano lavorare bene con gli altri è straordinario. L'alternativa dell'evangelizzazione/formazione/tirocinio vale la pena, ma è più costosa e richiede più tempo.

— CEO di un fornitore di software OSS

Ciò dimostra che l'assunzione di persone con un DNA open source è un percorso di accelerazione da considerare nella strategia delle risorse umane.

Processi

- Stabilire o rivedere le descrizioni delle mansioni (competenze tecniche, soft skill, competenze ed esperienze)
- Programmi di formazione: autoformazione, formazione formale, coaching manageriale, peer mapping, comunità
- Stabilire o rivedere il percorso di carriera: competenze, risultati/impatti chiave e fasi di carriera

Valutazione delle opportunità

1. Pratiche di sviluppo: il problema probabilmente non è tanto quello di spronare gli sviluppatori a usare più open source, quanto piuttosto quello di assicurarsi che lo usino in modo sicuro, nel rispetto dei termini di licenza di ogni tecnologia open source e senza abbandonare i tradizionali controlli di sicurezza (le linee di codice open source potrebbero contenere codice malevolo),
2. Rivedere le pratiche di collaborazione: con le pratiche di sviluppo, l'opportunità è quella di estendere l'agilità e la collaborazione ad altre linee di business dell'organizzazione. Per promuovere questi comportamenti si ricorre spesso all'inner sourcing, anche se questo potrebbe essere un mezzo passo avanti verso la cultura dell'open source,
3. Cultura dell'organizzazione: alla fine, tutto dipende dalla cultura della vostra organizzazione: l'open source può essere il fiore all'occhiello di valori come l'apertura, la collaborazione, l'etica, la sostenibilità.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Sono disponibili corsi di formazione per presentare sia i vantaggi che i vincoli (conformità ai termini di licenza della proprietà intellettuale) legati all'open source.
- Ogni sviluppatore, architect, responsabile di progetto (o Product Owner/Business Owner) comprende i vantaggi e i vincoli (conformità ai termini di licenza della proprietà intellettuale) legati all'open source.
- Gli sviluppatori sono incoraggiati a contribuire alle comunità open source e ad assumersene la responsabilità, e potrebbero ricevere una formazione adeguata per farlo.

- Le abilità e le competenze si riflettono nelle descrizioni delle mansioni e nelle fasi di carriera dell'organizzazione.
- L'esperienza acquisita dagli sviluppatori nell'open source (contributi alle comunità open source, partecipazione al processo di compliance interna, portavoce esterni dell'azienda, ...) viene presa in considerazione nel processo di valutazione delle risorse umane.

Strumenti

- Matrice di competenze.
- Programmi di formazione pubblica (es. scuola open source).
- Sourcing: GitHub, GitLab, LinkedIn, Meetup, Epitech, Epita...
- Modelli di contratto (clausola di fedeltà).
- Descrizioni di lavoro (modelli) e fasi di carriera (modelli).

Raccomandazioni

Nella maggior parte dei casi, oggi gli sviluppatori conoscono già alcuni principi dell'open source e sono disposti a lavorare con e sul software open source. Tuttavia, ci sono ancora alcune azioni che il management dovrebbe intraprendere:

- Preferenza per l'esperienza OSS nelle assunzioni, anche se il lavoro per cui lo sviluppatore viene assunto riguarda solo la tecnologia proprietaria. È probabile che, con la trasformazione digitale, lo sviluppatore debba un giorno lavorare sull'open source.
- Programma di formazione OSS: Ogni sviluppatore, ogni architetto, ogni responsabile di progetto (o Product Owner/Business Owner) dovrebbe avere accesso a risorse di formazione (video o formazione faccia a faccia) che presentino i vantaggi dell'open source e anche i vincoli in termini di proprietà intellettuale e conformità alle licenze.
- La formazione dovrebbe essere disponibile per gli sviluppatori che vogliono contribuire alle comunità open source e far parte degli organi di governance di queste comunità (certificazioni Linux).
- Riconoscimento, nei processi di valutazione personale delle risorse umane, del contributo del dipendente (sviluppatore o architetto) ai temi legati all'open source, come i contributi alle comunità open source e il rispetto dei termini di licenza della proprietà intellettuale. La maggior parte degli argomenti sono condivisi e rientrano nei percorsi di carriera tecnici, mentre alcuni potrebbero o dovrebbero essere specifici.
- Il segreto meglio custodito e la posizione dell'azienda: è necessario affrontare gli aspetti della comunicazione (quanto è importante per la vostra organizzazione, tanto da essere riflesso nel vostro rapporto annuale), come influisce sulla vostra posizione di comunicazione (un collaboratore open source potrebbe essere un portavoce della vostra azienda, compresi i contatti con la stampa).

Risorse

- Per quanto riguarda la capacità delle persone di parlare al di fuori dell'azienda durante gli eventi, si veda Activity 31: "(Obiettivo di coinvolgimento) Affermare pubblicamente l'uso dell'open source".

6.5 Upstream first

Activity ID: [GGI-A-39](#).

Descrizione

Questa attività si occupa di sviluppare la consapevolezza dei benefici del ritorno del contributo e di far rispettare il principio "upstream first".

Con l'approccio upstream first, tutto lo sviluppo di un progetto open source deve essere realizzato con il livello di qualità e apertura richiesto per essere sottoposto agli sviluppatori principali del progetto e da questi pubblicato.

Valutazione delle opportunità

Scrivere codice secondo l'approccio upstream first porta a:

- codice di migliore qualità,
- pronto per essere inviato in upstream,
- codice unito (merged) al software principale,
- compatibile con le versioni future,
- riconoscimento da parte della comunità di progetto e una collaborazione migliore e più proficua.

Upstream First è più di un semplice "essere gentili". Significa avere voce in capitolo nel progetto. Significa prevedibilità. Significa avere il controllo. Significa agire piuttosto che reagire. Significa capire l'open source. ([Maximilian Michels](#))

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi di questa attività: La fase a monte è stata implementata?

- Aumento significativo del numero di richieste di pull/merge inviate a progetti di terze parti.
- È stato redatto un elenco di progetti di terzi per i quali è necessario applicare il principio "upstream first".

Raccomandazioni

- Identificare gli sviluppatori con maggiore esperienza nell'interazione con gli sviluppatori upstream.
- Facilitare l'interazione tra sviluppatori e sviluppatori principali (eventi, hackathon, ecc.)

Risorse

- Una chiara spiegazione del principio Upstream First e del perché si inserisce nell'Obiettivo Cultura: <https://maximilianmichels.com/2021/upstream-first/>.

Upstream First significa che ogni volta che si risolve un problema nella propria copia del codice upstream da cui altri potrebbero trarre beneficio, si contribuiscono le modifiche a monte, cioè si invia una patch o si apre una richiesta di pull al repository upstream.

- [Cos'è l'Upstream e il Downstream nello sviluppo del software?](#) Una spiegazione chiarissima.
- Spiegato dai documenti di progettazione di Chromium OS: [Upstream First](#).
- Red Hat su upstream e i vantaggi di [upstream first](#).

7 Attività dell'obiettivo di coinvolgimento (Engagement Goal)

7.1 Impegnarsi in progetti open source

Activity ID: [GGI-A-29](#).

Descrizione

Questa attività consiste nell'impiegare contributi significativi ad alcuni progetti OSS che sono importanti per voi. I contributi sono scalati e impegnati a livello di organizzazione (non a livello personale come nell'Activity #26). Possono assumere diverse forme, dal finanziamento diretto all'allocazione di risorse (ad esempio, persone, server, infrastrutture, comunicazione, ecc.), purché vadano a beneficio del progetto o dell'ecosistema in modo sostenibile ed efficiente.

Questa attività è il seguito dell'Activity #26 e porta i contributi dei progetti open source al livello dell'organizzazione, rendendoli più visibili, potenti e vantaggiosi. In questa attività, si suppone che i contributi apportino un miglioramento sostanziale e a lungo termine al progetto OSS: ad esempio, uno sviluppatore o un team che sviluppa una nuova funzionalità molto richiesta, risorse infrastrutturali, server per un nuovo servizio, acquisizione della manutenzione di un ramo ampiamente utilizzato.

L'idea è quella di allocare una percentuale di risorse per sponsorizzare gli sviluppatori open source che scrivono e mantengono librerie o progetti che utilizziamo.

Questa attività implica una mappatura dei software open source utilizzati e una valutazione della loro criticità per decidere quale supportare.

Valutazione delle opportunità

Se ogni azienda che utilizza l'open source contribuisse almeno un po', avremmo un ecosistema sano. <https://news.ycombinator.com/item?id=25432248>

Il sostegno ai progetti aiuta a garantirne la sostenibilità e fornisce l'accesso alle informazioni, contribuendo forse anche a influenzare e a dare priorità ad alcuni sviluppi (anche se questo non dovrebbe essere il motivo principale del sostegno ai progetti).

Potenziamenti benefici di questa attività: garantire che le segnalazioni di bug siano prioritarie e che gli sviluppi siano integrati nella versione stabile. Possibili costi associati all'attività: impegno di tempo nei progetti, impegno di denaro.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Progetto beneficiario individuato.
- Opzione di supporto decisa, come il contributo monetario diretto o il contributo di codice.
- Nominato il task leader.
- È stato dato un contributo.
- Il risultato del contributo è stato valutato.

Punti di verifica presi in prestito dal questionario di OpenChain [autocertificazione](#):

- Abbiamo una politica di contributo a progetti open source per conto dell'organizzazione.
- Abbiamo una procedura documentata che regola i contributi open source.
- Abbiamo una procedura documentata per sensibilizzare tutto il personale del software alla politica di contribuzione open source.

Strumenti

Alcune organizzazioni offrono meccanismi di finanziamento di progetti open source (potrebbe essere conveniente se il vostro progetto target rientra nei loro portafogli).

- [Open Collective](#).
- [Software Freedom Conservancy](#).

- [Tidelift](#).

Raccomandazioni

- Concentratevi sui progetti critici per l'organizzazione: sono questi i progetti che desiderate maggiormente aiutare con il vostro contributo.
- Progetti comunitari mirati.
- Questa attività richiede un minimo di familiarità con il progetto target.

Risorse

- [Come sostenere i progetti open source ora](#): Una breve pagina con idee sul finanziamento dei progetti open source.
- [Sostenere l'OSS: uno spazio per conversazioni sul sostegno all'open source](#)

7.2 Sostenere le comunità open source

Activity ID: [GGI-A-30](#).

Descrizione

Questa attività consiste nel confrontarsi con i rappresentanti istituzionali del mondo open source.

Si ottiene attraverso:

- Aderire alle fondazioni OSS (compresi i costi finanziari dell'adesione).
- Attività di sostegno e promozione delle fondazioni.

Questa attività comporta l'assegnazione ai team di sviluppo e IT di tempo e budget per partecipare alle comunità open source.

Valutazione delle opportunità

Le comunità open source sono in prima linea nell'evoluzione dell'ecosistema open source. Impegnarsi con le comunità open source presenta diversi vantaggi:

- aiuta a tenersi informati e aggiornati,
- migliora il profilo dell'organizzazione,
- L'adesione comporta dei vantaggi,
- fornisce ulteriore struttura e motivazione al team IT open source.

I costi comprendono:

- quote associative,
- Il tempo del personale e il budget per le trasferte sono stati stanziati per partecipare alle attività della comunità,
- monitorare gli impegni in materia di proprietà intellettuale.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- L'organizzazione è membro firmatario di una fondazione open source.
- L'organizzazione partecipa alla governance.
- Il software sviluppato dall'organizzazione è stato inviato o aggiunto alla base di codice di una fondazione.
- L'adesione è riconosciuta sui siti web dell'organizzazione e della comunità.
- Esecuzione della valutazione costi/benefici dell'adesione.
- È stato nominato un punto di contatto per la comunità.

Raccomandazioni

- Entrate a far parte di una comunità compatibile con le vostre dimensioni e risorse, ossia una comunità che possa ascoltare la vostra voce e in cui possiate essere un contributo riconosciuto.

Risorse

- Date un'occhiata a questa [useful page](#) della Linux Foundation sul perché e come unirsi a una comunità open source.

7.3 Dichiarare pubblicamente l'uso dell'open source

Activity ID: [GGI-A-31](#).

Descrizione

Questa attività riguarda il riconoscimento dell'uso di OSS in un sistema informativo, in applicazioni e in nuovi prodotti.

- Fornire storie di successo.
- Presentare agli eventi.
- Finanziamento della partecipazione a eventi.

Valutazione delle opportunità

È ormai generalmente accettato che la maggior parte dei sistemi informativi si basa su OSS e che le nuove applicazioni sono in gran parte realizzate riutilizzando OSS.

Il vantaggio principale di questa attività è quello di creare condizioni di parità tra OSS e software proprietario, per far sì che l'OSS riceva la stessa attenzione e sia gestito in modo altrettanto professionale del software proprietario.

Un vantaggio secondario è che contribuisce ad aumentare il profilo dell'ecosistema OSS e, poiché gli utenti OSS sono identificati come "innovatori", aumenta anche l'attrattiva dell'organizzazione.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- I fornitori commerciali open source sono autorizzati a utilizzare il nome dell'organizzazione come riferimento del cliente.
- I collaboratori sono autorizzati a farlo e a esprimersi con il nome dell'organizzazione.
- L'uso degli OSS è apertamente menzionato nella relazione annuale del dipartimento IT.
- Non ci sono ostacoli al fatto che l'organizzazione spieghi l'uso degli OSS nei media (interviste, eventi OSS e di settore, ecc.).

Raccomandazioni

- L'obiettivo di questa attività non è che l'organizzazione diventi un organismo di attivismo OSS, ma che si assicuri che non ci siano ostacoli al riconoscimento dell'uso dell'OSS da parte del pubblico.

Risorse

- Esempio di [CERN](#) che afferma pubblicamente il proprio utilizzo di OpenStack

7.4 Impegnarsi con i fornitori open source

Activity ID: [GGI-A-33](#).

Descrizione

Assicuratevi i contratti con i fornitori di software open source che forniscono software di importanza cruciale per voi. Le aziende e le entità che producono software open source devono prosperare per garantire la manutenzione e lo sviluppo di nuove funzionalità. Le loro competenze specifiche sono necessarie per il progetto e la comunità di utenti fa affidamento sulla loro attività e sui loro contributi.

L'impegno con i fornitori open source assume diverse forme:

- Sottoscrivere piani di supporto.
- Appalto a società di servizi locali.
- Sponsorizzare gli sviluppi.
- Pagare per una licenza commerciale.

Questa attività implica la considerazione dei progetti open source come prodotti completi che valgono la pena di essere pagati, proprio come i prodotti proprietari, anche se di solito molto meno costosi.

Valutazione delle opportunità

L'obiettivo di questa attività è garantire un supporto professionale al software open source utilizzato nell'organizzazione. Presenta diversi vantaggi:

- Continuità del servizio grazie alla tempestiva correzione dei bug.
- Prestazioni di servizio grazie all'ottimizzazione dell'installazione.
- Chiarimento dello status legale/commerciale del software utilizzato.
- Accesso alle nuove informazioni.
- Previsioni di bilancio stabili.

Il costo è ovviamente quello dei piani di supporto scelti. Un altro costo potrebbe essere quello di abbandonare l'outsourcing di massa a grandi integratori di sistemi a favore di contratti di precisione con PMI esperte.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- L'open source utilizzato nell'organizzazione è sostenuto da un supporto commerciale.
- Sono stati stipulati piani di supporto per alcuni progetti open source.
- Il costo dei piani di supporto open source è una voce legittima del budget IT.

Raccomandazioni

- Quando possibile, trovare PMI esperte a livello locale.
- Diffidate dei grandi integratori di sistemi che rivendono competenze di terzi (rivendendo piani di assistenza che in realtà sono forniti da PMI esperte di open source).

Risorse

Un paio di link che illustrano la realtà commerciale del software open source:

- [Il punto di vista di un investitore sull'evoluzione della comunità verso il business dei progetti open source.](#)
- [Una lettura veloce per capire l'open source commerciale.](#)

7.5 Procurement policy open source

Activity ID: [GGI-A-43](#).

Descrizione

Questa attività riguarda l'implementazione di un processo di selezione, acquisizione e acquisto di software e servizi open source. Si tratta anche di considerare il costo effettivo del software open source e di provvedere alla sua fornitura. L'OSS può essere "gratuito" ad una prima analisi, ma non è esente da costi interni ed esterni come l'integrazione, la formazione, la manutenzione e il supporto.

Queste policy richiedono che sia le soluzioni open source che quelle proprietarie siano considerate simmetricamente quando si valuta il rapporto qualità-prezzo come la combinazione ottimale tra costo totale di proprietà e qualità. Pertanto, l'ufficio acquisti IT deve considerare attivamente ed equamente le opzioni open source, assicurando al contempo che le soluzioni proprietarie siano considerate sullo stesso livello nelle decisioni di acquisto.

La preferenza per l'open source può essere esplicitamente dichiarata sulla base della flessibilità intrinseca dell'opzione open source quando non c'è una significativa differenza del costo complessivo tra le soluzioni proprietarie e quelle open source.

I dipartimenti di approvvigionamento (Procurement departments) devono capire che le aziende che offrono supporto per gli OSS di solito non dispongono di risorse commerciali sufficienti per partecipare alle gare d'appalto e devono adattare di conseguenza le loro politiche e i loro processi di approvvigionamento per l'open source.

Valutazione delle opportunità

Diverse ragioni giustificano gli sforzi per istituire specifiche politiche di approvvigionamento dell'open source:

- L'offerta di software e servizi commerciali open source è in crescita e non può essere ignorata, e richiede l'implementazione di politiche e processi di approvvigionamento dedicati.
- Esiste una crescente offerta di soluzioni commerciali open source altamente competitive per i sistemi informativi aziendali.
- Anche dopo aver adottato un componente OSS gratuito e averlo integrato in un'applicazione, è necessario fornire risorse interne o esterne per la manutenzione del codice sorgente.
- Il costo totale di proprietà (TCO) è spesso (anche se non necessariamente) inferiore per le soluzioni FOSS: nessun costo di licenza da pagare al momento dell'acquisto/aggiornamento, mercato aperto per i fornitori di servizi, possibilità di fornire una parte o tutta la soluzione da soli.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Nuove Call for Proposals (proposte di nuove iniziative) richiedono in modo proattivo la presentazione di contributi open source.
- Il reparto acquisti ha un modo per valutare le soluzioni open source rispetto a quelle proprietarie.
- È stato implementato e documentato un processo di approvvigionamento semplificato per software e servizi open source.
- È stato definito e documentato un processo di approvazione che si avvale di competenze interfunzionali.

Raccomandazioni

- "Assicuratevi di attingere alle competenze dei vostri team IT, DevOps, cybersecurity, gestione del rischio e acquisti quando create il processo" (da [5 Open Source Procurement Best Practices](#)).
- La legge sulla concorrenza può richiedere che l'"open source" non sia specificamente menzionato.
- Selezionate la tecnologia in anticipo e poi affidatevi a RFP per la personalizzazione e i servizi di supporto.

Risorse

- [Fattori decisionali per l'acquisto di software open source](#): un po' datato, ma è comunque un'ottima lettura da parte dei nostri colleghi di OSS-watch nel Regno Unito. Date un'occhiata alle [slide](#).
- [5 Open Source Procurement Best Practices](#): un articolo recente sul procurement dell'open source con suggerimenti utili.

8 Attività dell'obiettivo strategico (Strategy Goal)

8.1 Impostare una strategia per la governance aziendale dell'open source

Activity ID: [GGI-A-16](#).

Descrizione

La definizione di una strategia di alto livello per la governance dell'open source all'interno dell'azienda assicura la coerenza e la visibilità degli approcci sia verso l'uso interno che verso i contributi e il coinvolgimento esterni. Rende più efficace la comunicazione aziendale offrendo una visione e una leadership chiare e consolidate.

Il passaggio all'open source comporta numerosi vantaggi, ma anche alcuni doveri e un cambiamento nella cultura aziendale. Può avere un impatto sui modelli di business e influenzare il modo in cui un'organizzazione presenta il suo valore e la sua offerta, nonché la sua posizione nei confronti dei clienti e dei concorrenti.

Questa attività comprende i seguenti compiti:

- Istituire un responsabile OSS, con la sponsorizzazione e il sostegno del top management.
- Stabilire e pubblicare una chiara tabella di marcia per l'open source, con obiettivi dichiarati e benefici attesi.
- Assicuratevi che tutto il management di alto livello ne sia a conoscenza e agisca di conseguenza.
- Promuovere l'OSS all'interno dell'azienda: incoraggiare le persone a usarlo, favorire le iniziative interne e il livello di conoscenza.
- Promuovere l'OSS all'esterno dell'azienda: attraverso dichiarazioni e comunicazioni ufficiali e un coinvolgimento visibile nelle iniziative OSS.

La definizione, la pubblicazione e l'applicazione di una strategia chiara e coerente favorisce il coinvolgimento di tutte le persone all'interno dell'azienda e facilita ulteriori iniziative da parte dei team.

Valutazione delle opportunità

È un buon momento per lavorare su questa attività se:

- Non c'è uno sforzo coordinato da parte del management e l'open source è ancora visto come una soluzione ad hoc.
- Esistono già iniziative interne, ma non penetrano fino ai livelli superiori della dirigenza.
- L'iniziativa è stata avviata da tempo, ma incontra molti ostacoli e non produce ancora i risultati attesi.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste una chiara carta di governance dell'open source per l'azienda. La carta dovrebbe contenere:
 - cosa ottenere,
 - per chi lo facciamo,
 - quale sia il potere degli strategist e cosa no.
- Una roadmap open source è ampiamente disponibile e accettata in tutta l'azienda.

Raccomandazioni

- Creare un gruppo di persone e processi per definire e monitorare la governance dell'open source all'interno dell'azienda. Assicuratevi che ci sia un chiaro impegno da parte del top management nei confronti delle iniziative open source.
- Condividere la strategia open source all'interno dell'organizzazione, renderla una questione importante e un vero impegno aziendale.

- Assicurarsi che la roadmap e la strategia siano ben comprese da tutti, dai team di sviluppo al management e al personale dell'infrastruttura.
- Comunicare i propri progressi, in modo che le persone sappiano a che punto è l'organizzazione rispetto al proprio impegno. Pubblicare regolarmente aggiornamenti e indicatori.

Risorse

- [Lista di controllo e riferimenti per l'Open Governance](#).
- [L'open source comme enjeu de souveraineté numérique, di Cédric Thomas, CEO di OW2, Workshop presso Orange Labs, Parigi, 28 gennaio 2020 \(solo in francese\)](#).
- [Una serie di guide per la gestione dell'open source in azienda, a cura della Linux Foundation](#).
- [Un bell'esempio di documento strategico open source, a cura del gruppo LF Energy](#)

8.2 Consapevolezza dei C-Level

Activity ID: [GGI-A-34](#).

Descrizione

L'iniziativa open source di un'organizzazione produrrà i suoi benefici strategici solo se sarà applicata ai livelli più alti, integrando il DNA open source nella strategia e nel lavoro interno dell'azienda. Questo impegno non può avvenire se i dirigenti di alto livello e il top management non ne fanno parte. La formazione e la mentalità dell'open source devono essere estese anche a coloro che determinano le politiche, le decisioni e la strategia generale, sia all'interno che all'esterno dell'azienda.

Questo impegno garantisce che i miglioramenti pratici, i cambiamenti di mentalità e le nuove iniziative siano accolti con un sostegno costante, benevolo e sostenibile da parte della gerarchia, che porta a una maggiore partecipazione dei lavoratori. Modella il modo in cui gli attori esterni vedono l'organizzazione, apportando benefici alla reputazione e all'ecosistema. È anche un mezzo per affermare l'iniziativa e i suoi benefici a medio e lungo termine.

Valutazione delle opportunità

Questa attività diventa essenziale se/quando:

- L'organizzazione ha fissato obiettivi globali relativi alla gestione dell'open source, ma fatica a raggiungerli. È improbabile che l'iniziativa possa raggiungere qualcosa senza una buona conoscenza e un chiaro impegno da parte dei dirigenti di livello superiore.
- L'iniziativa è già partita e sta facendo progressi, ma i livelli gerarchici superiori non la seguono adeguatamente.

Se tutto va bene, dovrebbe risultare evidente che l'utilizzo dell'open source, se non ad hoc, richiede un approccio coerente e ben ponderato, data la gamma di team e di cambiamenti culturali che può apportare.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste un ufficio o un funzionario di governance incaricato di definire una strategia open source uniforme in tutta l'azienda e di garantire che l'ambito di applicazione sia chiaro.
- Esiste un impegno chiaro e vincolante da parte della gerarchia nei confronti della strategia OSS.
- La gerarchia comunica in modo trasparente il proprio impegno nei confronti del programma.
- La gerarchia è disponibile per discutere del software open source. Può essere sollecitata e stimolata per i suoi impegni.
- Esiste un budget e un finanziamento adeguati per l'iniziativa.

Raccomandazioni

Esempi di azioni associate a questa attività sono:

- Condurre una formazione per demistificare l'OSS al top management.
- Ottenere un'approvazione esplicita e pratica per l'uso e la strategia OSS.
- Menzionare e sostenere esplicitamente il programma OSS nelle comunicazioni interne.
- Menzionare e sostenere esplicitamente il programma OSS nelle comunicazioni pubbliche.

Open source is a *strategic enabler* that embarks *enterprise culture*. What does this mean?

- L'open source può essere sfruttato come un meccanismo per interrompere i fornitori e ridurre i costi di acquisizione del software.
 - Should open source come under the purview of *Software Asset Managers* or *purchasing departments*?
- Open source licences enshrine the freedoms that deliver the benefits of open source, but they also carry *obligations*. If not met appropriately, responsibilities can create legal, commercial and image risks to an organisation.
 - Le condizioni di licenza consentiranno la visibilità di aree del codice che dovrebbero rimanere riservate?
 - Avrà un impatto sul portafoglio brevetti della mia organizzazione?
 - Come devono essere formati e supportati i team di progetto su questo tema?
- Contribuire a progetti open source esterni è il punto in cui risiede il maggior valore dell'open source.
 - In che modo la mia azienda dovrebbe incoraggiare (e monitorare) questo aspetto?
 - Come dovrebbero usare gli sviluppatori applicazioni come GitHub, GitLab, Slack, Discord, Telegram o qualsiasi altro strumento che i progetti open source utilizzano abitualmente?
 - L'open source può influire sulle politiche delle risorse umane dell'azienda?
- Naturalmente non si tratta solo di contribuire, che dire dei miei progetti open source?
 - Sono pronto a fare innovazione *open*?
 - Come gestiranno i miei progetti i contributi *in entrata*?
 - Devo impegnarmi per creare una community per un determinato progetto?
 - Come devo guidare la community, quale ruolo devono avere i suoi membri?
 - Sono pronto a cedere le decisioni sulla roadmap a una community?
 - L'open source può essere uno strumento prezioso per ridurre la "silo-isation" (tendenza alla creazione di silos, gruppi isolati) tra i team aziendali?
 - Devo gestire il trasferimento dell'open source da un'entità aziendale a un'altra?

8.3 Open source e sovranità digitale

Activity ID: [GGI-A-35](#).

Descrizione

La sovranità digitale può essere definita come

"Capacità e opportunità di individui e istituzioni di svolgere il proprio ruolo nel mondo digitale in modo indipendente, intenzionale e sicuro" — Centro di competenza per l'informatica pubblica, Germania

Per condurre correttamente la propria attività, qualsiasi entità deve fare affidamento su altri partner, servizi, prodotti e strumenti. Esaminare i legami e i vincoli di queste dipendenze permette all'organizzazione di valutare e controllare la propria dipendenza da fattori esterni, migliorando così la propria autonomia e resilienza.

Ad esempio, il vendor lock-in è un forte fattore di dipendenza che può ostacolare i processi e il valore aggiunto dell'organizzazione e, pertanto, deve essere evitato. L'open source è una delle vie d'uscita da questo blocco. L'open source svolge un ruolo significativo nella sovranità digitale, consentendo una maggiore scelta tra soluzioni, fornitori e integratori e un maggiore controllo sulle roadmap IT.

Va notato che la sovranità digitale non è una questione di fiducia: è ovvio che dobbiamo fidarci dei nostri partner e fornitori, ma il rapporto diventa ancora migliore quando si basa sul consenso

e sul riconoscimento reciproco, piuttosto che su contratti e costrizioni forzate.

Ecco alcuni vantaggi di una migliore sovranità digitale:

- Migliorare la capacità dell'organizzazione di fare le proprie scelte senza vincoli.
- Migliorare la resilienza dell'azienda nei confronti di attori e fattori esterni.
- Migliorare la posizione negoziale nei rapporti con partner e fornitori di servizi.

Valutazione delle opportunità

- Quanto è difficile/costoso abbandonare una soluzione?
- I fornitori di soluzioni potrebbero imporre condizioni indesiderate sul loro servizio (ad esempio, cambio di licenza, aggiornamento dei contratti)?
- I fornitori di soluzioni potrebbero aumentare unilateralmente i loro prezzi, semplicemente perché non abbiamo scelta?

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Esiste una valutazione delle dipendenze critiche per i fornitori e i partner dell'organizzazione.
- Esiste un piano di backup per queste dipendenze identificate.
- Esiste un requisito dichiarato di sovranità digitale quando si studiano nuove soluzioni.

Raccomandazioni

- Identificare i principali rischi di dipendenza dai fornitori di servizi e dalle entità terze.
- Mantenere un elenco di alternative open-source ai servizi critici.
- Aggiungere un requisito nella selezione di nuovi strumenti e servizi utilizzati all'interno dell'ente, affermando la necessità di sovranità digitale.

Risorse

- [A Primer on Digital Sovereignty & Open Source: part I](#) e [A Primer on Digital Sovereignty & Open Source: part II](#), dal sito Open-Sourcerers.
- Un eccellente articolo di [superuser.openstack.org](#) su [The Role of Open Source in Digital Sovereignty](#). Ecco un breve estratto:
La sovranità digitale è una preoccupazione fondamentale per il XXI secolo, soprattutto per l'Europa. L'open source ha un ruolo importante nel consentire la sovranità digitale, permettendo a tutti di accedere alla tecnologia necessaria, ma anche fornendo la trasparenza della governance e l'interoperabilità necessarie per il successo di queste soluzioni.
- Il punto di vista dell'Unione Europea sulla sovranità digitale, dall'[Osservatorio Open Source \(OSOR\)](#): Open Source, sovranità digitale e interoperabilità: La dichiarazione di Berlino.
- La posizione dell'UNICEF su [Open Source for Digital Sovereignty](#).

8.4 L'open source che consente l'innovazione

Activity ID: [GGI-A-36](#).

Descrizione

L'innovazione è l'attuazione pratica di idee che portano all'introduzione di nuovi beni o servizi o al miglioramento dell'offerta di beni o servizi.

— Schumpeter, Joseph A.

L'open source può essere un fattore chiave per l'innovazione grazie alla diversità, alla collaborazione e allo scambio fluente di idee. Persone con background e ambiti diversi possono avere prospettive diverse e fornire risposte nuove, migliori o addirittura dirompenti a problemi noti. Si può favorire l'innovazione ascoltando i diversi punti di vista e promuovendo attivamente una collaborazione aperta su progetti e argomenti.

Allo stesso modo, partecipare all'elaborazione e all'implementazione di standard aperti è un grande promotore di buone pratiche e idee per migliorare il lavoro quotidiano dell'azienda. Inoltre, consente all'azienda di guidare e influenzare l'innovazione identificando cosa le serve e in quale fase, aumentando la sua visibilità e reputazione a livello globale.

Attraverso l'innovazione, l'open source permette non solo di trasformare i beni o i servizi che la vostra azienda commercializza, ma anche di creare o modificare l'intero ecosistema in cui la vostra azienda vuole prosperare.

Ad esempio, rilasciando Android come open source, Google invita centinaia di migliaia di aziende a creare i propri servizi basati su questa tecnologia open source. Google sta quindi creando un intero ecosistema di cui tutti i partecipanti possono beneficiare. Naturalmente, poche aziende sono abbastanza potenti da creare un ecosistema con la propria decisione. Ma ci sono molti esempi di alleanze tra aziende per creare un tale ecosistema.

Valutazione delle opportunità

È importante valutare la posizione della vostra azienda rispetto ai concorrenti, ai partner e ai clienti, perché spesso sarebbe rischioso per un'azienda allontanarsi troppo dagli standard e dalle tecnologie utilizzate da clienti, partner e concorrenti. Innovare significa ovviamente essere diversi, ma ciò che si differenzia non deve avere una portata troppo ampia; in caso contrario, l'azienda non potrebbe beneficiare degli sviluppi software realizzati dalle altre aziende dell'ecosistema e dello slancio commerciale che l'ecosistema offre.

Valutazione dei progressi

I seguenti **punti di verifica** dimostrano i progressi in questa attività:

- Sono state identificate le tecnologie - e le comunità open source che le sviluppano - che hanno un impatto sull'azienda.
- I progressi e le pubblicazioni di queste comunità open source sono monitorati - sono persino a conoscenza della loro strategia prima che i rilasci siano resi pubblici.
- I dipendenti dell'organizzazione sono membri di (alcune) di queste comunità open source e ne influenzano le roadmap e le scelte tecniche contribuendo con linee di codice e partecipando agli organi di governance di queste comunità.

Raccomandazioni

Tra tutte le tecnologie necessarie per la gestione della vostra azienda, dovrete individuarne alcune:

- le tecnologie che potrebbero essere le stesse dei vostri concorrenti,
- le tecnologie che devono essere specifiche per la vostra azienda.

Rimane aggiornati sulle tecnologie emergenti. L'open source ha guidato l'innovazione nell'ultimo decennio e molti strumenti potenti di giorno in giorno provengono da lì (si pensi a Docker, Kubernetes, ai progetti Apache Big Data o a Linux). Non è necessario sapere tutto su tutto, ma si dovrebbe conoscere abbastanza lo stato dell'arte per identificare nuove tendenze interessanti.

Consentite e incoraggiate le persone a presentare idee innovative e a portarle avanti. Se possibile, destinare risorse a queste iniziative e farle crescere. Affidatevi alla passione e alla volontà delle persone di creare e promuovere idee e tendenze emergenti.

Risorse

- [4 innovazioni che dobbiamo all'open source.](#)
- [The Innovations of Open Source](#), del professor Dirk Riehle.
- [Open source technology, enabling innovation.](#)
- [Can Open Source Innovation Work in the Enterprise?.](#)
- [Europe: Open source software strategy.](#)
- [Europe: Open source software strategy 2020-2023.](#)

8.5 L'open source che consente la trasformazione digitale

Activity ID: [GGI-A-37](#).

Descrizione

“La trasformazione digitale è l'adozione della tecnologia digitale per trasformare i servizi o le aziende, sostituendo processi non digitali o manuali con processi digitali o sostituendo la vecchia tecnologia digitale con una più recente” (Wikipedia)

Quando le organizzazioni più avanzate nella Trasformazione Digitale guidano congiuntamente il cambiamento attraverso il Business, l'IT e la Finanza per ancorare il digitale nel percorso, riesaminano:

- Modello di business: catena del valore con ecosistemi, as a service, SaaS.
- Finanza: opex/capex, persone, outsourcing.
- IT: innovazione, modernizzazione di legacy/asset.

L'open source è al centro della trasformazione digitale:

- Tecnologie, pratiche agili, gestione dei prodotti.
- Persone: collaborazione, comunicazione aperta, ciclo di sviluppo/decisione.
- Modelli di business: try & buy, open innovation.

In termini di competitività, i processi più visibili sono probabilmente quelli che hanno un impatto diretto sulla customer experience. E dobbiamo riconoscere che i grandi player, così come le start-up, offrendo un'esperienza del tutto inedita ai clienti, hanno cambiato drasticamente le loro aspettative.

L'esperienza del cliente e tutti gli altri processi di un'azienda dipendono interamente dall'IT. Ogni azienda deve trasformare il proprio IT, questo è il senso della trasformazione digitale. Le aziende che non l'hanno ancora fatto devono realizzare la loro trasformazione digitale il più velocemente possibile, altrimenti il rischio è di essere spazzate via dal mercato. La trasformazione digitale è una condizione di sopravvivenza. Poiché la posta in gioco è così alta, un'azienda non può affidare completamente la trasformazione digitale a un fornitore. Ogni azienda deve mettere mano all'IT, il che significa che ogni azienda deve mettere mano al software open source, perché non esiste IT senza software open source.

I vantaggi attesi dalla trasformazione digitale includono:

- Semplificare, automatizzare i processi principali e renderli in tempo reale.
- Consentire risposte rapide ai cambiamenti della concorrenza.
- Sfruttare l'intelligenza artificiale e i big data.

Valutazione delle opportunità

La trasformazione digitale potrebbe essere gestita da:

- Segmenti dell'IT: IT di produzione, IT di supporto al business (CRM, fatturazione, acquisti...), IT di supporto (HR, finanza, contabilità...), Big Data.
- Tipo di tecnologia o processo a supporto dell'IT: infrastruttura (cloud), intelligenza artificiale, processi (Make-or-Buy, DevSecOps, SaaS).

L'introduzione dell'open source in un particolare segmento o tecnologia dell'IT rivela che si vuole mettere mano a questo segmento o tecnologia, perché si è valutato che questo particolare segmento o tecnologia dell'IT è importante per la competitività dell'azienda. È importante valutare la posizione della vostra azienda rispetto non solo ai vostri concorrenti, ma anche ad altri settori e ai principali attori in termini di esperienza del cliente e soluzioni di mercato.

Valutazione dei progressi

1. Livello 1: valutazione della situazione

- Ho identificato:
 - i segmenti IT che sono importanti per la competitività della mia azienda, e
 - le tecnologie open source necessarie per sviluppare applicazioni in questi segmenti.

- E così ho deciso:
 - su quali segmenti voglio gestire internamente lo sviluppo dei progetti, e
 - su quali tecnologie open source devo costruire le mie competenze interne.

1. Livello 2: coinvolgimento

- Su alcune tecnologie open source selezionate e utilizzate in azienda, diversi sviluppatori sono stati formati e sono riconosciuti come validi collaboratori dalla comunità open source.

In alcuni segmenti selezionati sono stati avviati progetti basati su tecnologie open source.

1. Livello 3: Generalizzazione

- Per tutti i progetti, un'alternativa open source viene sistematicamente studiata durante la fase iniziale del progetto. Per facilitare lo studio di tali alternative open source da parte del team di progetto, un budget centrale e un team centrale di architetti, ospitati nel Dipartimento IT, sono dedicati a fornire assistenza ai progetti.

KPI:

- KPI 1. Rapporto per il quale è stata esaminata un'alternativa open source: (Numero di progetti / Numero totale di progetti).
- KPI 2. Rapporto per il quale è stata scelta l'alternativa open source: (Numero di progetti / Numero totale di progetti).

Raccomandazioni

Al di là dei titoli, la Trasformazione Digitale è una mentalità che implica alcuni cambiamenti fondamentali, che dovrebbero provenire anche (o addirittura soprattutto) dai livelli più alti dell'organizzazione. Il management deve promuovere iniziative e nuove idee, gestire i rischi e potenzialmente aggiornare le procedure esistenti per adattare ai nuovi concetti.

La passione è un enorme fattore di successo. Uno dei mezzi sviluppati dai principali attori del settore è la creazione di spazi aperti per le nuove idee, dove le persone possono presentare e lavorare liberamente sulle loro idee di trasformazione digitale. Il management dovrebbe incoraggiare queste iniziative.

Risorse

- [Eclipse Foundation: Enabling Digital Transformation in Europe Through Global Open Source Collaboration.](#)
- [Europe: Open source software strategy.](#)
- [Europe: Open source software strategy 2020-2023.](#)

9 Conclusioni

Come abbiamo già detto, la Good Governance dell'open source non è una destinazione, ma un viaggio. Dobbiamo prenderci cura dei nostri beni comuni, delle comunità e dell'ecosistema che li fanno prosperare, perché da questo dipende il nostro successo comune, e quindi individuale.

Noi, in qualità di professionisti del software e appassionati di open source, ci impegniamo a continuare a migliorare il manuale della Good Governance Initiative e a lavorare sulla sua diffusione e portata. Crediamo fermamente che le organizzazioni, gli individui e le comunità debbano lavorare fianco a fianco per costruire un insieme di beni comuni migliore e più ampio, disponibile e vantaggioso per tutti.

Voi siete invitati a unirvi alla OSPO Alliance, a contribuire al nostro lavoro, a spargere la voce e a farvi ambasciatori di una migliore consapevolezza e governance dell'open source all'interno del vostro ecosistema. Le risorse disponibili sono numerose, dai post sui blog agli articoli di ricerca, dalle conferenze ai corsi di formazione online. Anche noi mettiamo a disposizione una serie di materiali utili sul [nostro sito web](#), e siamo felici di poter dare il nostro contributo per quanto possibile.

Definiamo e costruiamo insieme il futuro della Good Governance Initiative!

9.1 Contatti

Il modo migliore per entrare in contatto con la OSPO Alliance è quello di inviare un messaggio alla nostra mailing list pubblica all'indirizzo <https://accounts.eclipse.org/mailling-list/ospo.zone>. Potete anche venire a discutere con noi ai consueti eventi open source, partecipare ai nostri webinar mensili OSPO OnRamp o mettervi in contatto con un qualsiasi membro, che vi indirizzerà gentilmente alla persona giusta.

9.2 Appendice: Modello di Customised Activity Scorecard

L'ultima versione del modello di Customised Activity Scorecard è disponibile nella sezione resources del [Good Governance Initiative GitLab](#) di OW2.

Iniziativa OW2 OSS Good Governance		THE GOOD EXAMPLE COMPANY		Customised Activity Scorecards	
Obiettivo/Attività Culture 1		Promuovere le migliori pratiche di sviluppo open source		Ultimo aggiornamento 07/28/21	
Descrizione personalizzata <i>Ambito di ciò che deve essere fatto</i> Breve descrizione essenziale... <ul style="list-style-type: none">Brevi punti salienti...			Valutazione delle opportunità <i>Perché questa attività è rilevante</i> <ul style="list-style-type: none">Punti critici principali...Opportunità di avanzamento chiave...		
Obiettivi <i>Cosa vogliamo ottenere in questa iterazione</i> <ul style="list-style-type: none">Obiettivo 1...Obiettivo 2...		Strumenti <i>Tecnologie, strumenti e prodotti utilizzati nell'attività</i> <ul style="list-style-type: none">Risorse...		Note operative <i>Approccio, metodo per progredire nell'Activity</i> <ul style="list-style-type: none">Iniziare con...	
Risultato chiave <i>Come misureremo il successo di questa iterazione</i>			Progressi	Punteggio	Valutazione personale
1. Risultato chiave 1 (minimo un risultato chiave)			xx%	.9	Commento personale
2. Risultato chiave 2			xx%	.5	Commento personale
3. Risultato chiave 2			xx%	.5	Commento personale
4. Risultato chiave 4 (massimo quattro risultati chiave)			xx%	.0	Commento personale
				.475	
Timeline <i>Date di inizio e fine, Milestones</i> <ul style="list-style-type: none">Indicazione della data qui		Effort <i>Budget di tempo e materiale</i> <ul style="list-style-type: none">Tempo a disposizione per i prossimi tre mesiAllocazione del budget		Assegnatari <i>Chi partecipa? Chi guida?</i> <ul style="list-style-type: none">XX per preparare la presentazione interna	
Issues <i>Difficoltà, incertezze, ostacoli, punti di attenzione, dipendenze</i> <ul style="list-style-type: none">Problema 1...Problema 2...			Stato <i>Come sta andando l'attività</i> Commento personale sullo stato dell'Activity		
			Valutazione complessiva dei		XX%

Iniziativa OW2 OSS Good Governance	THE GOOD EXAMPLE COMPANY	Customised Activity Scorecards
------------------------------------	---------------------------------	--------------------------------

Note

Approfondimenti dal forum delle Activity di GitLab

<https://gitlab.ow2.org/ggi/ggi-castalia/-/issues/25>

Copiare/incollare qui il contenuto della descrizione dell'Activity da <https://gitlab.ow2.org/ggi/ggi-castalia/>
Questo servirà come riferimento per sviluppare la Customized Activity Scorecard