

Sviluppo di un Blog Decentralizzato Tramite Tecnologia SOLID

Laurea in ingegneria informatica ed elettronica

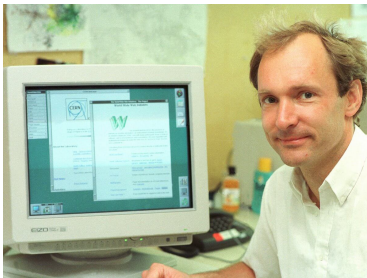
Candidato: *Carlo Tosoni*
Relatore: *Luca Grilli*

24/9/2021

Contenuti

- 1 Il Problema Affrontato
- 2 Requisiti
- 3 Architettura del sistema
- 4 Esempi di utilizzo
- 5 Conclusioni

Perché nacque il Web



6 agosto 1991: invenzione del World Wide Web.

“Ho immaginato il Web come una piattaforma aperta che avrebbe permesso a tutti, ovunque, di condividere informazioni, di accedere ad opportunità e di collaborare attraverso differenti aree geografiche e confini culturali”

Sir Tim Berners-Lee.

Problemi del Web secondo Sir Tim Berners-Lee

Three challenges for the Web, according to its inventor

- ❶ We've lost control of our personal data
- ❷ It's too easy for misinformation to spread on the Web
- ❸ Political advertising online needs transparency and understanding

Problemi relativi alla centralizzazione dei dati

Quando i dati sono salvati in piattaforme digitali chiuse e centralizzate:

- ❶ Nessuna visibilità su ciò che viene conservato
- ❷ Scarso controllo su come i dati vengono utilizzati
- ❸ Non si può scegliere quali applicazioni usare per accedervi
- ❹ Non si possono usare i dati come unità coesiva

=> Nascita di una nuova tecnologia per ridecentralizzare il Web.

Solid



Progetto voluto da Tim Berners-Lee e sostenuto da molti esperti di Sicurezza Informatica e del Semantic Web.

Obiettivo: ridecentralizzare il World Wide Web limitando il monopolio dei dati dell'utente.

Concetti chiave relativi alla tecnologia Solid

Solid permette agli utenti di salvare i propri dati in stores decentralizzati chiamati **Pod**. Ogni utente può decidere quali altre persone e applicazioni possano accedervi, concedendo loro, o eventualmente revocando loro, diritti su una risorsa/e contenuta/e all'interno del **Pod**.

Possibilità di concedere diversi livelli di permesso: **Read, Append, Write, Control**.

Concetti chiave relativi alla tecnologia Solid

Termini legati alla tecnologia Solid

- ❶ **Pod**: Luogo per salvare dati dell'utente
- ❷ **webId**: Identificatore univoco utente
- ❸ **Solid Identity Provider**: Compagnia che permette di autenticarsi e ospitare Pod
- ❹ **Container**: "Directory" all'interno del Pod
- ❺ **SolidDataset**: Dataset contenente informazioni appartenenti all'utente

Obiettivi della tesi

1) Creazione di un'applicazione completamente decentralizzata in accordo con la tecnologia Solid.

Per permettere ad utenti di creare/gestire un proprio blog personale.

L'applicazione è stata denominata **my-solid-blog**.

Obiettivi della tesi

2) Dimostrare l'utilità e l'importanza di tale tecnologia.

Alcuni vantaggi:

- Controllo dei propri dati
- Arginato problema della disinformazione
- Evitare forme di censura

Creata applicazione blog-validator per controllare autenticità dei dati dell'utente.

Il sistema SAdEB

Tesi finalizzata allo sviluppo di due applicazioni,
my-solid-blog e **blog-validator**, facenti parte del sistema
denominato **SAdEB**.

Candidato: *Carlo Tosoni* **Relatore:** *Luca Grilli*

Dati contenuti nel Pod



```

@prefix : "<#>".
@prefix acl: <http://www.w3.org/ns/auth/acl#>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix ldap: <http://www.w3.org/ns/ldap#>.
@prefix schema: <http://schema.org/>.
@prefix solid: <http://www.w3.org/ns/solid/terms#>.
@prefix space: <http://www.w3.org/ns/pim/space#>.
@prefix vcard: <http://www.w3.org/2006/vcard/ns#>.
@prefix pro: <./>.
@prefix inbox: </inbox/>.
@prefix c: </>.

pro:card a foaf:PersonalProfileDocument; foaf:maker :me; foaf:primaryTopic :me.

:me
  a schema:Person, foaf:Person;
  vcard:fn "Carlo";
  vcard:organization-name "Università di Perugia";
  vcard:role "Student";
  acl:trustedApp
  [
    acl:mode acl:Append, acl:Control, acl:Read, acl:Write;
    acl:origin <https://podbrowser.inrupt.com>
  ],
  [
    acl:mode acl:Append, acl:Read, acl:Write;
    acl:origin <http://localhost:3000>
  ],
  [
    acl:mode acl:Append, acl:Read, acl:Write;
    acl:origin <http://localhost:3001>
  ],
  [
    acl:mode acl:Append, acl:Read, acl:Write;
    acl:origin <http://localhost:8081>
  ],
  [
    acl:mode acl:Append, acl:Read, acl:Write;
    acl:origin <https://notepod.vincenttunzu.com>
  ];
  ldap:inbox inbox::;
  space:preferenceFile </settings/prefs.ttl>;
  space:storage c::;
  solid:account c::;
  solid:privateTypeIndex </settings/privateTypeIndex.ttl>;
  solid:publicTypeIndex </settings/publicTypeIndex.ttl>;
  foaf:name "Carlo".

```

Requisiti

Requisiti delle applicazioni facenti parte del sistema **SADeB**.

my-solid-blog:

- Applicazione completamente decentralizzata
- Possibilità di creare/gestire il proprio blog
- Permettere di leggere blog di altri utenti (inserendo webld proprietario)
- Verificare autenticità contenuti postati

Requisiti

Requisiti delle applicazioni facenti parte del sistema **SADeB**.

blog-validator:

- Server per controllare autenticità dei contenuti postati
- Possibilità per l'utente di verificare personalmente esito controllo

Architettura del sistema

Tecnologie utilizzate per lo sviluppo delle due applicazioni:

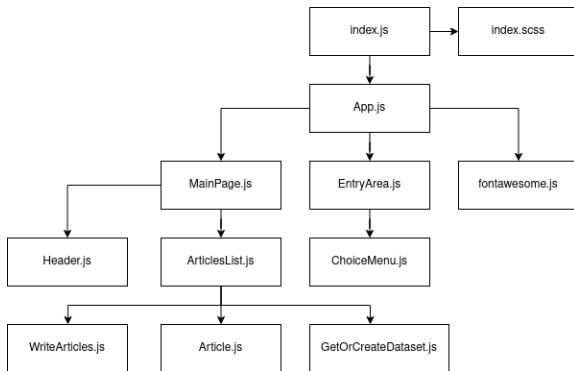
my-solid-blog

- ❶ **Inrupt**: Librerie JavaScript create dalla società Inrupt, fondata da Berners-Lee per gestire dati all'interno del Pod
- ❷ **React**: Libreria open-source, front-end, JavaScript
- ❸ **Bulma**: Framework CSS open-source

blog-validator

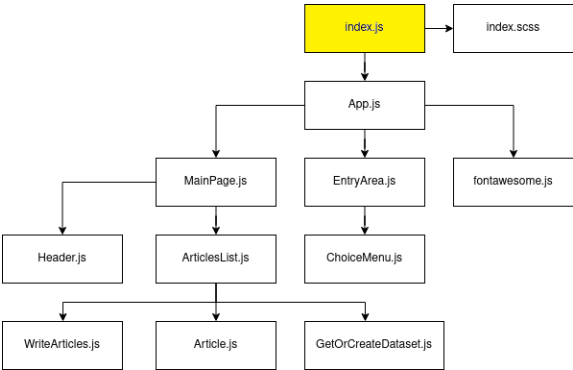
- ❶ **Node.js**: Runtime system multiplatforma orientato agli eventi

my-solid-blog



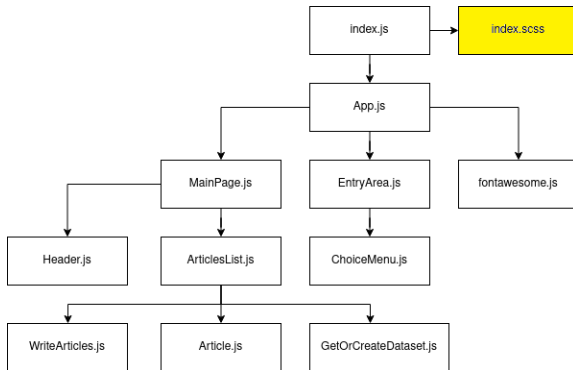
L'albero delle dipendenze dei moduli di **my-solid-blog**.

my-solid-blog



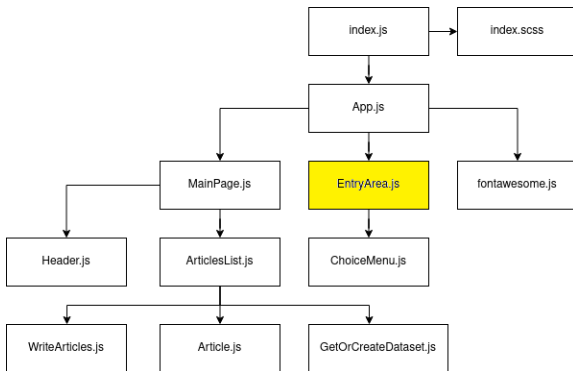
index.js: Entry point di JavaScript.

my-solid-blog



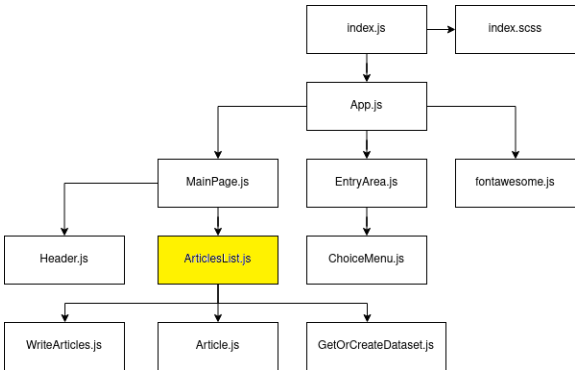
index.scss: Importa componenti del framework Bulma per la formattazione del documento HTML.

my-solid-blog



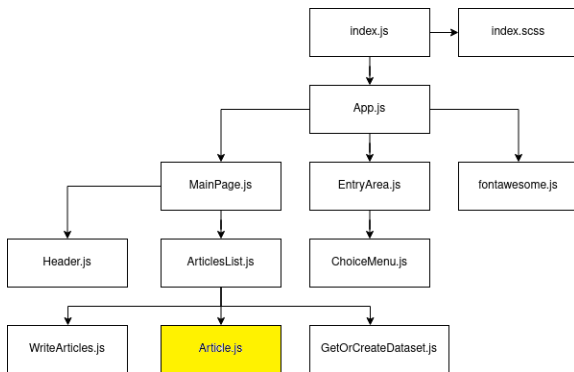
EntryArea.js: Renderizza schermata prima di effettuare l'accesso. Permette autenticazione con **Solid-Identity-Provider**, oppure inserimento della **webId** del proprietario del blog.

my-solid-blog



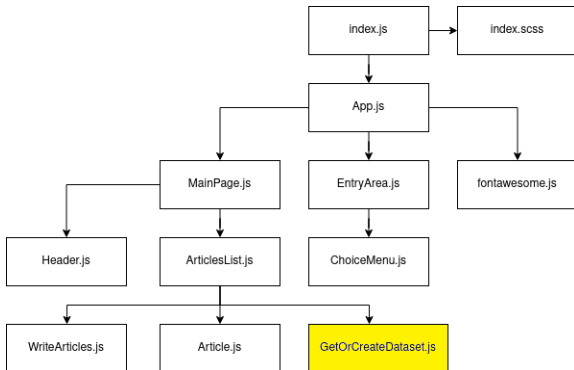
Articlelist.js: Si occupa di renderizzare il contenuto di un blog e di elaborare le informazioni contenute all'interno del **SolidDataset** ove è salvato.

my-solid-blog



Article.js: Renderizza il contenuto dei singoli articoli del blog.

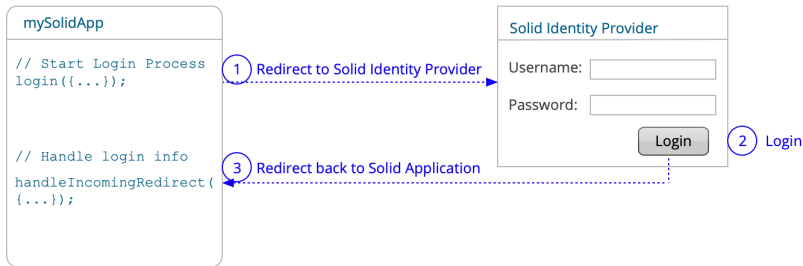
my-solid-blog



GetOrCreateDataset.js: Si occupa dell'interazione con il **Pod** per quanto concerne il caricamento del **SolidDataset** contenente il blog dell'utente. Qualora tale **SolidDataset** non esista, lo inizializza.

my-solid-blog

Autenticazione con il Solid-Identity-Provider



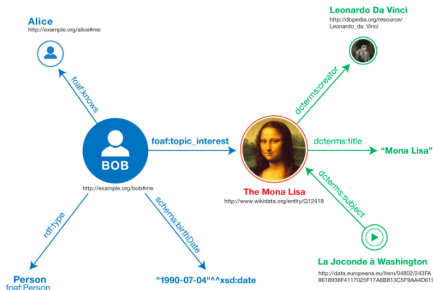
Autenticazione con il proprio Solid-Identity-Provider implementata tramite libreria **Inrupt** solid-ui-react.

my-solid-blog

Gestione dei dati all'interno del Pod

Ogni applicazione deve potenzialmente accedere ai dati contenuti all'interno del Pod. Necessaria modalità unica per la rappresentazione dei dati. Utilizzato linguaggio **RDF**.

Resource Description Framework



Qualsiasi dato in **RDF** è una risorsa rappresentata tramite un **IRI**, pertanto è reperibile direttamente dal Web. Utilizzati diversi vocabolari per la rappresentazione di differenti tipologie di dati.

Resource Description Framework

Importanti vocabolari RDF

- ❶ **RDF**: Vocabolario fondamentale, definisce un modello per la rappresentazione dati in RDF
- ❷ **FOAF**: Per rappresentare persone, organizzazioni e interazioni sociali
- ❸ **ACL**: Vocabolario per la gestione dei livelli di accesso di un agente su una risorsa
- ❹ **Schema**: Definisce varie strutture di dati

Resource Description Framework

Linked Data

RDF rappresenta tutte le informazioni tramite **Statements**.
Statements formati da: **Soggetto**, **Predicato** e **Oggetto**.

`< https : // carlotosoni99.inrupt.net/profile/card#me >` sogg.
`< http : //xmlns.com/foaf/0.1/name >` pred.
`"Carlo"` ogg.

my-solid-blog

Utilizzata libreria **Inrupt solid-client** per gestione dati nel Pod. Contenuto blog rappresentato da un **SolidDataset** chiamato **articlelist.ttl**. Ogni articolo del blog è un oggetto **RDF**.

- ❶ Caricamento **SolidDataset** dal **Container** ove è contenuto
- ❷ Creazione/Modifica/Rimozione di uno o più oggetti **RDF** contenuti nel **SolidDataset**
- ❸ Salvataggio del nuovo **SolidDataset** all'interno del **Pod**

my-solid-blog

Codice per il caricamento del SolidDataset.

```
export async function getOrCreateDataset(containerUri, fetch, publicTypeIndexUri) {  
  
  const publicTypeIndex = await getSolidDataset(publicTypeIndexUri);  
  
  const articleListThing = getThing(publicTypeIndex, "https://carlotosoni99.inrupt.net/settings/publicTypeIndex.ttl#articlelist");  
  
  if ( articleListThing === null ) {  
    try{  
      const articlelistUri = containerUri + "articlelist.ttl";  
      return await initialiseArticlelist(articlelistUri, fetch, publicTypeIndex, publicTypeIndexUri);  
    }  
    catch(e) {  
      console.log("an error occurr")  
      console.log(e);  
    }  
  }  
  
  const articlelistUri = getUrl(articleListThing, solid.instance);  
  
  return await getSolidDataset(articlelistUri);  
}
```

my-solid-blog

I singoli articoli contenuti all'interno del blog sono stati definiti tramite il vocabolario **Schema**. Ogni articolo è un oggetto **RDF** appartenente alla classe **TextDigitalDocument**.

Proprietà utilizzate:

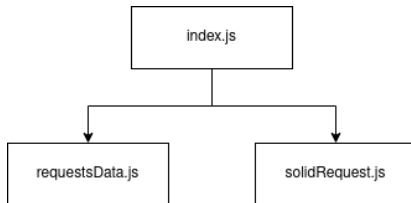
- **dataCreated**: Data creazione articolo
- **headline**: titolo articolo
- **text**: contenuto articolo
- **identifier**: identificatore univoco articolo

blog-validator

Per evitare diffusione di disinformazione o forme di censura da parte di my-solid-blog, è stata creata una seconda applicazione, esterna alla prima, chiamata **blog-validator**.

blog-validator effettua dei controlli sull'autenticità dei contenuti mostrati su richiesta dell'utente.

blog-validator



L'albero delle dipendenze dei moduli di **blog-validator**.

blog-validator

I moduli di blog-validator

- **index.js**: Entry point di JavaScript, definisce le **API methods** necessarie per effettuare controlli sull'autenticità dei dati mostrati da **my-solid-blog**, e per permettere all'utente di verificare personalmente l'esito di tali controlli
- **solidRequest.js**: Legge i dati relativi al blog all'interno del **Pod** dell'utente, per confrontarli con quelli ricevuti da **my-solid-blog**
- **requestsData.js**: Memorizza in un database locale i dati relativi alle richieste di controllo dei dati mostrati da **my-solid-blog** per visualizzarli all'utente qualora questo ne richieda la visione, per la gestione del database utilizza la libreria **sqlite3**

Comunicazione tra le due applicazioni

La comunicazione tra **my-solid-blog** e **blog-validator** avviene tramite protocollo **HTTP**. Premendo il pulsante "Check", **my-solid-blog** invia una richiesta **POST** al server **blog-validator**, contenente tutti i parametri necessari per effettuare un controllo sui dati.

Parametri passati a blog-validator per il controllo

- ❶ **webId**: WebId del proprietario dell'articolo
- ❷ **urlDataset**: URL del **SolidDataset** contenente l'articolo
- ❸ **urlThing**: URL dell'articolo
- ❹ **title**: Titolo dell'articolo
- ❺ **content**: Contenuto dell'articolo
- ❻ **date**: Data di creazione dell'articolo

Usata libreria **axios** per stabilire comunicazione **HTTP**.

my-solid-blog utilizza la porta **3000**, mentre **blog-validator** la porta **8081**.

Avvio my-solid-blog

Welcome to my-solid-blog

You can use this application by logging in with your Solid Identity Provider or by entering a valid webid of a user (without any login process)

Log in

Enter WebId

[read article's url](#)

8/3/2021 19:14

  Check

8/3/2021 19:13

  Check

This application was developed for the thesis of Carlo Tosoni (a.a. 2020/2021), bachelor degree in computer engineering.

Possibilità modifica contenuto articolo

Sed ut perspiciatis unde omnis iste
natus error sit voluptatem accusantium
doloremque laudantium, totam rem
aperiam.

aque ipsa quae ab illo inventore
veritatis et quasi architecto beatae
vitae dicta sunt explicabo. Nemo enim
ipsam voluptatem quia voluptas sit
aspernatur aut odit aut fugit, sed quia
consequuntur magni dolores eos qui
ratione voluptatem sequi nesciunt.
Neque porro quisquam est, qui
dolorem ipsum quia dolor sit amet,
consectetur, adipisci velit, sed quia
non numquam eius modi tempora
incididunt ut labore et dolore magnam
aliquam quaerat voluptatem. Ut enim
ad minima veniam, quis nostrum
exercitationem ullam corporis suscipit

Cancel

Save

Contenuto blog in RDF







▼ **articlelist.ttl**




```

@prefix : <#>.
@prefix schema: <http://schema.org/>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

:16280108379165575986189264108
  a "http://schema.org/TextDigitalDocument";
  schema:dateCreated "2021-08-03T17:13:57.922Z"^^xsd:dateTime;
  schema:headline
    "Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque
    laudantium, totam rem aperiam,";
  schema:identifier 0.7234417465494506;
  schema:text
    "aque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt
    explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia
    consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui
    dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi
    tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam,
    quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi
    consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil
    molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?".
:16314558109764018660049026021
  a "http://schema.org/TextDigitalDocument";
  schema:dateCreated "2021-09-12T14:10:10.980Z"^^xsd:dateTime;
  schema:headline "Lorem ipsum dolor sit amet, consectetur adipisci elit,";
  schema:identifier 0.7957625451495327;
  schema:text
    "sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis
    nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi
    consequatur. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
    pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit
    anim id est laborum.".

```


Controllo autenticità articolo

[read article's url](#)

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam,

8/3/2021 19:13

aque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?



Check

Article of <https://carlotosoni99.inrupt.net/profile/card#me> found, all in order

blog-validator

Blog Validator

Use the input below to see which data the application "my-solid-blog" sent for the control

You have to enter the url of the article checked

<https://carlotosoni99.inrupt.net/public/my-solid-blog/articlelist.ttl#1628010876907897671766654623>

Use the inputs below to search the content of an article

Enter here the SolidDataset's url

Dataset's url here

Enter here the Article's url

Article's url here

Then press "check"

check

Conclusioni e sviluppi futuri

Solid, pur essendo una tecnologia dal grande potenziale, è ancora in via di sviluppo. Pertanto alcune sue funzionalità non sono state ancora completamente definite, oppure non presentano un'adeguata documentazione.

A tal proposito, per esempio, nell'implementazione dell'applicazione **my-solid-blog**, vi sono stati alcuni problemi riguardanti la gestione delle risorse **ACL**, essenziali per la gestione dei livelli di accesso di un agente sulle risorse contenute all'interno del Pod.

Conclusioni e sviluppi futuri

“The Web Access Control specification is not yet finalised. As such, this function is still experimental and subject to change, even in a non-major release.”

Frase presente all'interno della documentazione **Inrupt**,
relativa alle funzioni per la gestione delle risorse **ACL**.

Conclusioni e sviluppi futuri

Solid è una tecnologia concepita per arginare il fenomeno della eccessiva centralizzazione del Web, i dati di milioni di utenti, ottenuti in cambio di servizi gratuiti, sono in pratica "imprigionati" nei data center dai grandi giganti operanti in questo settore.

La tecnologia **Solid** può sicuramente mitigare questo fenomeno, ponendo le basi per lo sviluppo di piattaforme digitali ove in controllo dei dati è restituito agli utenti, riducendo il rischio di fenomeni di censura e favorendo lo sviluppo di meccanismi di validazione delle informazioni.

Conclusioni e sviluppi futuri

Scandalo Facebook-Cambridge Analytica (2018):

Raccolta dei dati di 87 milioni di utenti Facebook, senza il loro consenso, per scopi di propaganda politica.

Forte necessità di modificare il Web al fine di decentralizzarlo.

Ringrazio tutti per l'attenzione