

DaanKG: An Ontology model of the UN Sustainable Development Goals to Facilitate and Improve Corporate Social Responsibility

Michael DeBellis¹, Cara Arellano², Patrick Guo³, Tejas Jyothi⁴, Vishnu Suresh⁵, and Kenneth Kron⁶

¹ michaeldebellis.com, San Francisco, CA, USA

² CEO DaanMatch PBC, Oakland, CA, USA

³ CDO DaanMatch PBC, Hayward, CA, USA

^{4,5} University of California, Berkeley, USA

⁶ CISO DaanMatch PBC, Moonpark, CA, USA

Abstract

The DaanMatch project seeks to: 1) Enable community based Non-Governmental Organizations (NGOs) to meet administrative requirements for government mandated Corporate Social Responsibility (CSR) programs 2) Utilize big data to help donors intelligently target India's development goals using the framework provided by the United Nation's Sustainable Development Goals (SDGs) 3) Utilize advanced technology to increase the reach and impact of funding for global development. The initial research has been carried out in the context of India. India is a good test bed due to their ambitious government mandated CSR program. This paper describes one module of the DaanMatch system developed with ontology and knowledge graph technology: OWL, SPARQL, Protégé, Cellfie, the AllegroGraph graph database product and Python APIs from Franz Inc. The DaanMatch Knowledge Graph module (DaanKG) provides an ontology that models the UN Sustainable Development Goals, NGOs, and CSR programs as well as the records, people, and deliverables associated with NGO projects. The DaanMatch system demonstrates a new paradigm that utilizes technology to radically reinvent the funding and monitoring of projects enabling more focus on helping those in need rather than on administrative overhead.

Keywords

Ontology, Web Ontology Language, United Nations, Sustainable Development Goals, Knowledge Graph, India

1. Introduction: The DaanMatch Vision

India suffers from staggering inequality.² Despite steady economic growth the gap between those with and those without has continued to expand. While India's richest 1% hold over 40% share of the nation's wealth, the half of the population at the lower end controls just 6%. [1] Non-profit organizations work to provide for those in need but struggle for resources. With the introduction of Section 135 of the Companies Act, 2013, India became the first country to mandate Corporate Social Responsibility (CSR). This legislation was intended to provide much-needed resources to Non-Government Organizations (NGOs) to compensate for dwindling government programs. [2] Ten years after going into effect, the mandate has not lived up to India's aspirations. [3]

OSS2023: *Ontologies for Services and Society, 9th Joint Ontology Workshops (JOWO 2023), co-located with FOIS 2023, 19-20 July, 2023, Sherbrooke, Québec, Canada*

EMAIL: mdebellisf@gmail.com (A. 1); cara.arellano@berkeley.edu (A. 2); shpatrickguo@gmail.com (A. 3)

ORCID: 0000-0002-8824-9577 (A. 1); 0000-0003-3642-8634 (A. 2); 0009-0008-6434-3754 (A.3)



© 2023 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

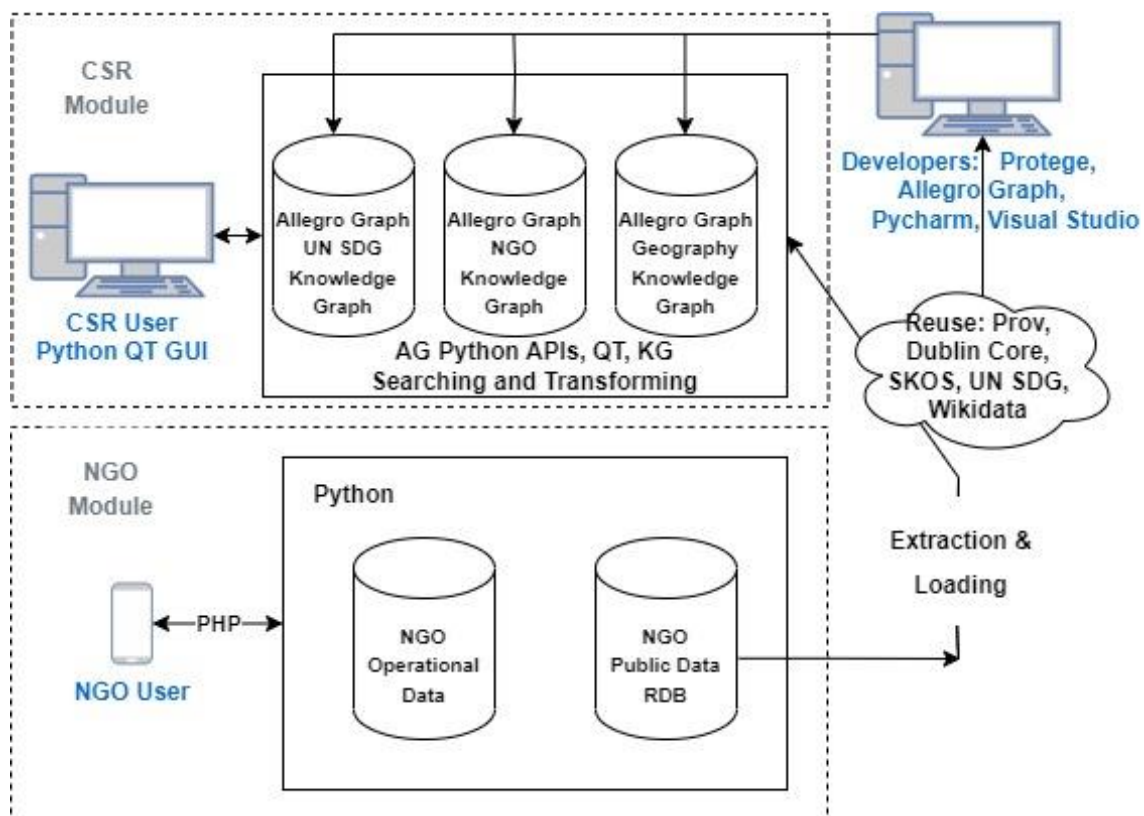


Figure 1: The DaanMatch System Architecture

Annual CSR spending for 2020-21 was approximately \$3.1 Billion USD. In 2020-21 37% of Indian corporations were non-compliant. Almost half of all CSR funding is allocated to NGOs in urban areas in the state of Maharashtra and in Delhi, with the lion's share going to a small number of large organizations, while scores of small, effective, local NGOs, in regions with the most need are unable to access funding. [3] Smaller, local NGOs providing programs and services in the country's most impoverished regions, often lack manpower, resources, and basic business skills, needed to complete the current application and administration processes. [3] [4]

Compared to large NGOs that incur costs of travel, accommodations, high-priced consultants, and related expenses, grassroots NGOs have lower operating costs. [4] [5] Smaller, local NGOs are often run and manned by volunteers and spend more of their budget on direct services, to help those in need. In addition, because they tend to have close ties to the local culture, they often can provide better services for those in need. Despite this, the current system greatly favors larger NGOs, especially for corporate CSR programs that need to limit the time they spend finding, funding, and monitoring NGOs. [4]

The Companies Act is an admirable attempt to direct funding to NGOs. In practice, however, challenges faced by the corporations, the NGOs, and the government, have restricted program effectiveness overall, left billions of dollars in available aid untouched, and have left a vast number of effective grassroots NGOs neglected. [3] [6] [7]

DaanMatch is a system designed to address these issues utilizing technologies such as machine learning and knowledge graphs. DaanMatch seeks to democratize transparency and ease the burden of compliance for both Donors and NGOs, making formal funding easier and more equitable. The goals of DaanMatch are to:

1. Increase the pool of NGOs that are eligible for funding by helping NGOs use their data to show transparency and impact easily, using evidence in lieu of narrative and paperwork.

2. Save donors time and money by helping them connect with projects that address sustainable development goals, align with donor values, and are implemented by legitimate, effective organizations.
3. Provide better metrics on how effective CSR programs are in achieving the UN Sustainable Development Goals.

The DaanMatch system consists of two main modules (see Figure 1): the CSR module, is focused on CSR managers who wish to analyze data about NGOs and how they relate to SDGs in specific areas. The NGO module utilizes technologies such as speech to text, video capture, image capture, and metadata to simplify the process for NGOs to receive and demonstrate compliance with CSR grants. As a result, the NGO module must run on mobile devices. Many NGO users in rural areas, especially smaller NGOs, use their phones as their only platform to access the Internet and applications. In addition, the vision of DaanMatch: to utilize technology to radically simplify the compliance process, requires that NGO users have their devices with them when they perform the services stipulated in project grants. The solution to obstacles facing NGOs were revealed during an unrelated volunteer opportunity with the Human Rights Center Lab at UC Berkeley. Monitoring events at middle east hotspots on social media for potential human rights violations, the project used metadata, reverse image and satellite imagery to remotely evaluate and substantiate claims. We theorized that similar methods could be used to verify work done by NGOs. I.e., video activities such as food distribution, sanitation development, etc. Our idea is to automate these processes for NGO data collection and apply these technologies to social impact and global development, streamlining the effort for transparency and easing reporting for grassroots NGOs.

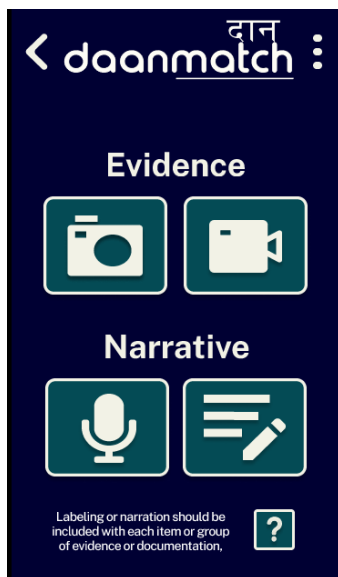


Figure 2: NGO Mobile GUI

As a result, the NGO module requires a web-based client and more conventional technology that will enable NGO users to easily record appropriate media and upload it into the DaanMatch database with the appropriate metadata. Figure 2 shows a screen from the NGO mobile GUI. NGO users can use this to document activities by taking videos, recording audio, using speech to text, etc. The information is automatically tagged with appropriate metadata indicating where, when, and by whom the media was captured and linking it with the appropriate project (grant) that the event was associated with. The CSR module on the other hand, is divided into two different distributions:

1. An open-source distribution that will be freely available to all interested users. This is the version that is the short-term focus.
2. A proprietary distribution for corporations that utilize DaanMatch to manage their CSR program. This is the long-term vision and will include data specific to each NGO and corporation that utilizes DaanMatch to fund and monitor projects. This data will include the project status, the SDGs that it addresses, the media that document how the NGO fulfilled the requirements of the project, etc.

Distribution 1 provides an open-source platform for all users to browse the data collected by the DaanMatch team about NGOs. This data has been “scraped” from various public web sites such as GuideStar.[7] In addition, it is supplemented with Linked Data³ about geographic locations in India as well as other public data that may be useful to CSR managers, CSR executives, UN workers, economists, and academic researchers. Distribution 2 will utilize all the knowledge in distribution 1 supplemented with specific data for each CSR program. This module is more appropriate for complex models that are best supported by formal ontologies and knowledge graphs. This module is more recent and is built on an ontology that models the United Nations Sustainable Development Goals (SDG). This is the module that will be the focus of this paper.

³ As of the publication date: July, 2023, Linked Data has not yet been integrated into the system. See section 4 for more detail.

Section 2 describes the various ontologies and tools used to develop the knowledge graph system. It includes specific examples of how the formal ontology model can facilitate the DaanMatch goals. Section 3 describes the development process for this module. Section 4 discusses future plans and how our experience reflects on the use of formal ontologies for real world data and problems.

2. The DaanMatch CSR Module

This section will describe the DaanMatch CSR module and how it is designed around formal ontologies. The project is taking an Agile [8] approach to development. The current ontology and code can be found at: https://github.com/mdebellis/Daan_Knowledge_Graph.

2.1. Integrating the Knowledge Graph with the NGO Database Model

The data model for NGOs and other tables utilized in the NGO module are replicated as data properties in the CSR ontology. Super properties are utilized to organize the data properties so that it is easy to manage changes made to the relational data model and to upload the data from the relational database into the knowledge graph. For example, there are tables in the relational database called NGOBackground, NGOContact, and NGOFinance. The various columns associated with each table are replicated as data properties in the ontology and are made sub-properties of the super-properties: ngoBackgroundProperty, ngoContactProperty and ngoFinanceProperty. Figure 3 shows a screen capture from the Protégé ontology editor with some of these super-properties expanded.

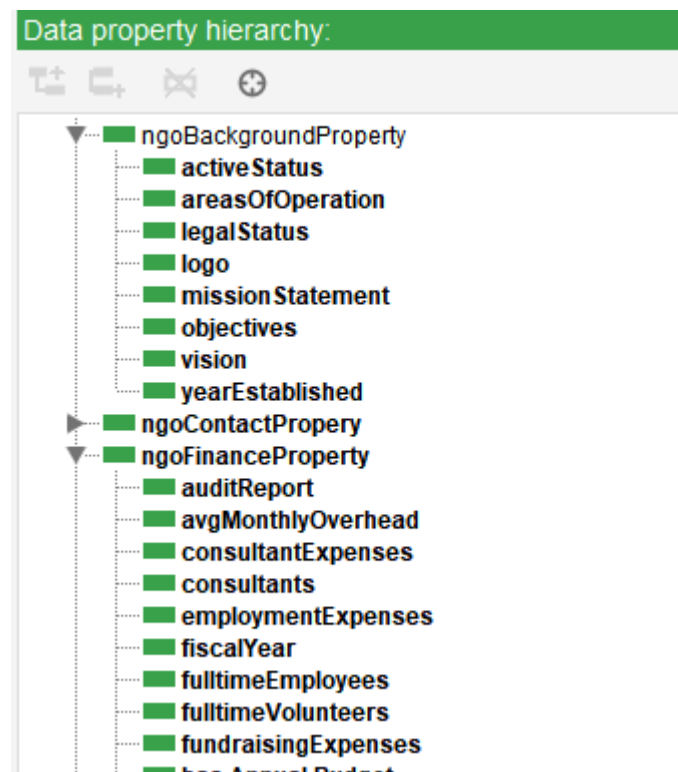


Figure 3: Data Super-properties for DBMS Integration

this are:

- Uploading the data with minimal transformation makes the upload process less error prone and more maintainable.
- Uploading data with minimal transformation provides an audit trail showing where information in the ontology originated. This is especially important for our problem because our vision is that recording media and metadata that records specific instances of

This makes the uploading of data from the relational database very straight forward. The domain for each of the sub-properties is defined on the super-property and inherited by each sub-property. In some cases, such as NGOFinance, there is a class that corresponds to the table. In other cases, such as NGOContact, the data in the database (name, phone, email) applies to other classes in the ontology that are not modeled in the database. In those cases, the domain for the super-property is a higher level class than the table such as the Prov:Agent class for NGOContact.

After the data has been loaded from the database it is transformed into a knowledge graph format.

This inverts the standard Extract, Transform, and Load (ETL) paradigm into Extract, Load, and Transform (ELT). [9] This is becoming common in many uses of ontologies for real data. The reasons for

Prov:Activity will significantly reduce and eventually eliminate standard bureaucratic paperwork. As part of transforming the data, the Python functions will correct datatype and other errors as much as possible. At the same time, correcting such data must be flagged for potential auditing because it may indicate fraud or errors in the data.

- Transforming data that is in the knowledge graph allows the transformations to take advantage of axioms in the ontology and the reasoner. This simplifies the transformation process and allows more sophisticated kinds of transformations. [10]

Additional knowledge that is not required for the NGO module is added from ontologies based on the UN Sustainable Development Goals model and linked data sources such as Wikidata. All 3 ontologies were initially modeled in the Protégé ontology editor. In order to support the large amount of data required for the ontologies, the data is stored in a knowledge graph hosted in the AllegroGraph product from Franz Inc. Python is used to:

- Perform batch uploads from the NGO module into the knowledge graph.
- Transform the database data into a knowledge graph format. I.e., go from “string to things”. [11]
- Perform various manipulations and analysis of the data. E.g., link NGOs and CSRs to appropriate UN SDGs via text matching.
- Present the information in the ontologies to CSR users in a GUI.

Grouping properties together in this way makes maintenance and testing significantly easier when managing large amounts of data. The same technique was used in [10] and several consulting projects that the lead author has done for industry clients. An example of the usefulness of this approach is that when the team realized that many of our classes should be replaced with classes from the Prov-O vocabulary [12] having the domains of these classes specified on super-properties made this change significantly easier than if they were defined on each data property. This is an example of how formal models can facilitate an Agile development process.

2.2. Ontology Design and Reuse

In this section we describe the 3 main ontologies in the CSR module. In the current system these are all represented in one ontology and the boundaries between the (eventually separate) ontologies is defined by different namespaces such as Prov, SDG, and NGO. The ontologies are currently in one large ontology because as we were developing, we were making large changes to the design on a regular basis, and it was easier to make such changes with the entire model in one ontology. Now that the model has mostly been finalized, we will divide the large ontology into 3 different ontologies and different AllegroGraph repositories. This is discussed in section 3.

2.2.1. The UN Sustainable Development Goals Ontology

The most important metrics for describing and analyzing data about NGO and CSR programs comes from the United Nations Sustainable Development Goals. [13] The UN has a large amount of data available to describe these goals on their SDG portal. [14] The SDG ontology primarily consists of 3 classes:

1. SDGGoal. These are the 17 high level goals defined by the UN. For example:⁴ *Goal 1: End poverty in all its forms everywhere.*
2. SDGTarget. Each goal has one or more targets. The SDG goals are high-level aspirational goals. The SDG targets are more concrete goals that relate to each high-level goal. For example, a target for Goal 1 is: *Target 1.1: By 2030, eradicate extreme poverty for all people everywhere, currently measured as people living on less than \$1.25 a day.*

⁴ <https://sdgs.un.org/goals/goal1>

3. **SDGIndicator.** Each target has one or more indicators. These are concrete metrics that measure how well a nation or region is achieving the target. For example, the indicator for target 1.1 is: *Indicator 1.1.1: Proportion of the population living below the international poverty line by sex, age, employment status and geographical location (urban/rural).*

Due to the well thought out design and numbering of each goal, target, and indicator and the meta data provided by the UN in a spreadsheet on their SDG data portal it was extremely easy to model the SDGs in OWL. The process to develop this ontology was to:

1. Rework the metadata spreadsheet so that it was in the proper format to be uploaded by the Cellfie plugin [15] for Protégé.
2. Use Cellfie to create the appropriate instances of each of the classes described above. In addition, metadata that described the identifier for each Goal, Target, and Indicator was added as an annotation property values (modeled via Dublin Core) for each new instance. E.g., the identifier for Goal 1 is the string “1.” The identifier for Target 1.1 is “1.1” and the identifier for Indicator 1.1.1 is “1.1.1”.
3. Use SPARQL to utilize the identifier keys to automatically create appropriate object property links between each Goal, Target, and Indicator. For example, the SPARQL code to create the hasIndicator values from a target to its indicator is shown in Figure 4. This code processes each SDGGoal and finds any instances of SDGTarget that begin with the same identifier as the goal. If the target has the same prefix as the goal, then the goal hasTarget of the target. E.g., it will find that the identifier for target 1.1 begins with the identifier for goal 1 and hence will add a property value illustrating that the goal hasTarget for that target.

```
CONSTRUCT {?g :has_Target ?t}
WHERE {?g a :SDG_Goal;
       :identifier ?gi.
       ?t a :SDG_Target;
       :identifier ?ti.
       BIND (CONCAT(?gi, ".") AS ?gid)
       BIND (IF (STRSTARTS(?ti, ?gid), "Match", "NoMatch") AS ?match)
       FILTER (?match = "Match")}
```

Figure 4: Example SPARQL Query to add hasTarget property values

The UNSDG ontology was developed before this project and is available as a separate ontology at [16].

2.2.2. Geography Ontology

The geography of India includes states, territories, cities, wards, towns, and villages. Rather than re-creating this information manually, we chose to utilize Linked Data [17] and download the appropriate classes and instances from Wikidata. This is a large enough amount of data of its own that it merits its own ontology. In addition to the basic geographic information such as latitude, longitude, and population, we hope to incorporate additional information from Wikidata and other linked data sources that will further allow users to perform sophisticated analysis of the CSR data in relation to the UN SDG metrics as well as specific metrics that may relate to each corporation’s CSR vision.

2.2.3. The NGO Ontology

The NGO Ontology has the majority of the classes. These are classes to model NGOs, Corporations and their CSR programs, grants that have been allocated from a CSR to an NGO, media (Prov:Entity) that are used to document the fulfillment of a grant, and other miscellaneous classes and properties that describe the NGO/CSR process and data.

We considered using an Upper Model; however, we did not see any tangible benefit from using any of the popular Upper Models. In addition, there was significant reuse potential in the Prov-O ontology. Prov-O provides: "...the foundation to implement provenance applications in different domains that can represent, exchange, and integrate provenance information generated in different systems and under different contexts". [12] This is directly relevant to our long term goals of using media (instances of the Prov:Entity class) to supplement and eventually mostly replace traditional documentation for NGO projects.

The two most significant ways that knowledge graph technology has been utilized are to:

1. Use text matching to align NGOs and CSRs with the UN goals that they are most focused on. Then utilize these links to the goals to match NGOs with CSR programs.
2. Create graphs that document how instances of Prov:Entity such as videos document the work done by an NGO to fulfill the goals of a specific project.

These directly map to the goals discussed in the introduction. Alignment of CSRs and NGOs with the UN SDGs simplifies the process of CSRs finding NGOs that align with their vision. The knowledge graph connecting NGOs to Prov:Entities created on site and tagged with metadata helps to achieve the goal of radically simplifying the CSR funding process so that smaller, less computer literate NGOs will be able to show compliance with project grant requirements with minimal effort. These will be discussed next in sections 2.3 and 2.4.

2.3. Linking NGOs and CSRs via Shared SDGs

?sdg	?csr	?ngo
Good Health And Well Being	HDFC Bank	Bankrishnapur Janas
Good Health And Well Being	ICICI Bank	Bankrishnapur Janas
Good Health And Well Being	Infosys	Bankrishnapur Janas
Good Health And Well Being	Kotak Mahindra Bank	Bankrishnapur Janas
Good Health And Well Being	Larsen Toubro Limited	Bankrishnapur Janas
Good Health And Well Being	Reliance Industries Limited	Bankrishnapur Janas
Good Health And Well Being	State Bank of India	Bankrishnapur Janas
Good Health And Well Being	Tata Consultancy Services	Bankrishnapur Janas
Gender Equality	HDFC Bank	Apne Aap Women's C

Figure 5: SPARQL query Utilizing SDGs to match CSRs and NGOs

precomputing indices for one or more triples. It allows intelligent search that can match words taking into account plurals, tense, synonyms, etc. It also allows specification of “stop words” (e.g., “a”, “and”, “the”) that are ignored because they are irrelevant to the domain.

We created an FTI called NGOIndex. This FTI indexes the values of data properties with strings that describe properties such as the NGO’s vision, description, and mission. In addition, we manually annotated each SDGGoal with keywords that are specific to that goal. E.g., a keyword for the NoPoverty goal is the string “poverty”. We then iterate through each FTI for each NGO and if there is an FTI match between the keyword for a goal and the NGO we create a new object property value linking the NGO with the SDG. We do the same matching for data properties that describe the CSR vision for a corporation. This allows us to do SPARQL queries that match CSR programs with NGOs.

One of the most important additions that the ontology provides to the data uploaded from the NGO module is a property: relatesToSDG with domain Prov:Organization and range sdg:SDGGoal. These links are created automatically utilizing an NLP feature in AllegroGraph called a Free Text Index (FTI). [18]

An FTI is created to provide various types of fast search by

Figure 5 shows the simple SPARQL query that can match CSR programs and NGOs that have compatible SDG visions. This is a screen print from the AllegroGraph Gruff SPARQL query and visualization tool. Gruff can take the results of a SPARQL query and automatically create a graph that

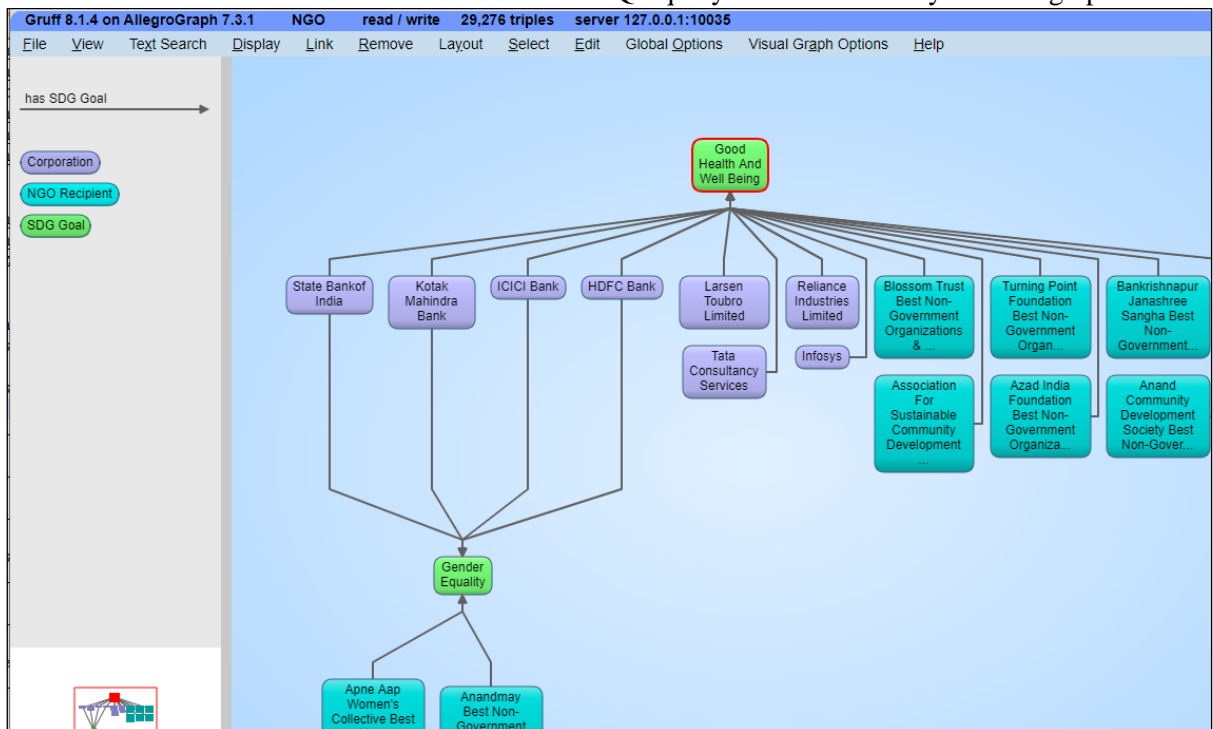


Figure 6: Gruff visualization of some results from query in figure 5

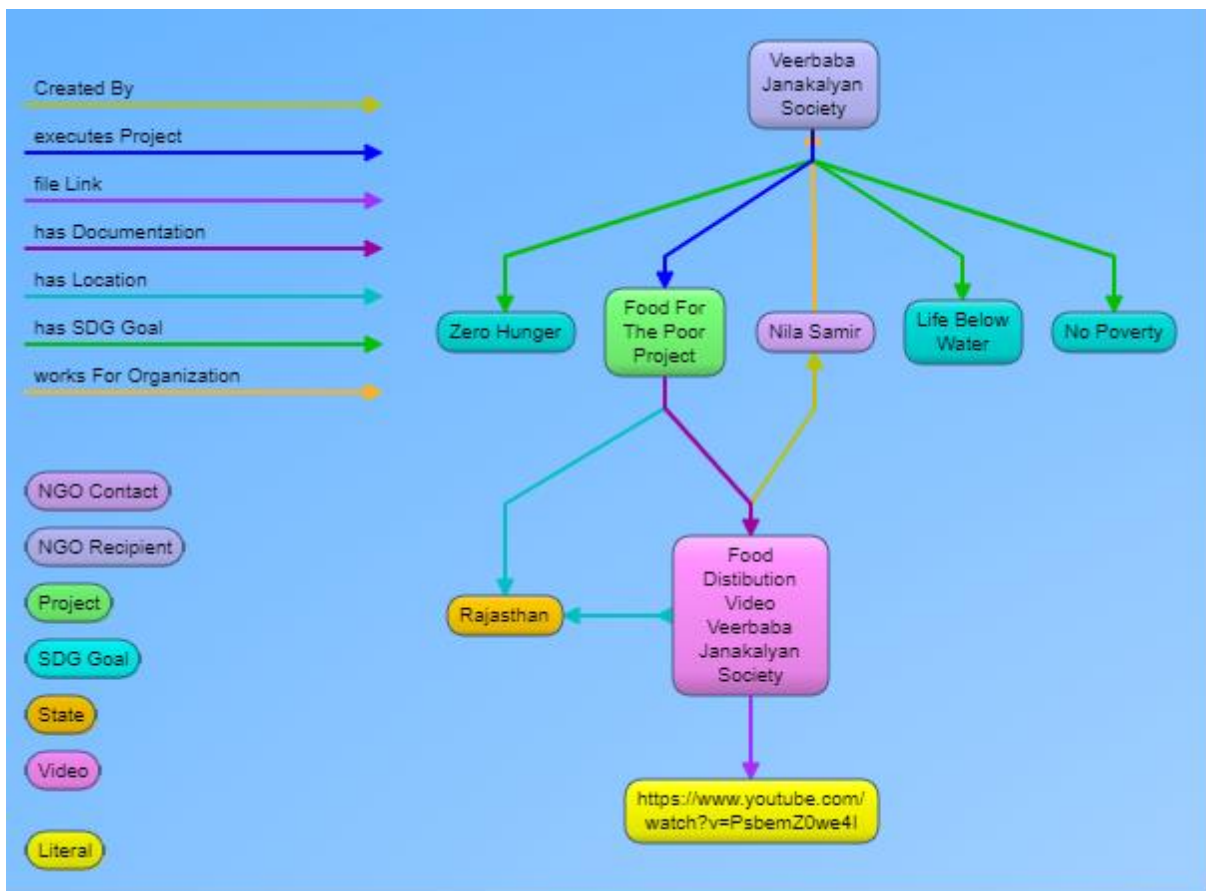


Figure 7: Visualization of individuals and property values documenting an NGO project

represents them. [19] In addition, the user can rework the graph in various ways such as selecting a specific node and having Gruff automatically generate a tree graph from that node.

Figure 5 is a screen print from Gruff displaying a visualization for some of the data in the SPARQL query. The key for the graph is in the left pane. Nodes are instances of classes and links are data and object properties with the name of the class or property color coded in the key on the left. In figure 6 purple nodes are instances of the Corporation class, light blue nodes are instances of the NGO class, and green nodes are instances of the SDGGoal class. There is only one link in figure 6, the hasSDGGoal object property color coded in black.

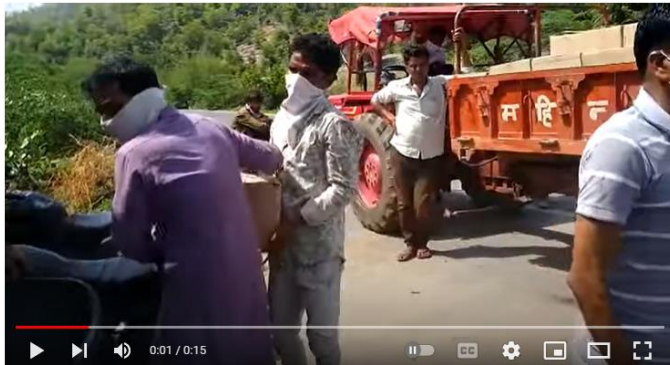


Figure 8: Video documenting project modeled in figure 7

Figure 7 is another screen print from Gruff, this time showing individuals and property values for a specific food distribution project that DaanMatch helped to facilitate. This graph includes the NGO, the location, and a video documenting the project (which in this case is available on YouTube). The thumbnail for this video is shown in figure 8.

2.4 Auditing Projects

An essential requirement to realizing the DaanMatch vision of using technology to significantly reduce the administrative burden on NGOs is to maintain the same or higher levels of verification with the new process.

In addition to capturing the relations among media used to document projects, the knowledge graph can facilitate auditing by modeling logic associated with the proper execution of a project. Such logic can be captured in axioms for relevant classes, rules in the Semantic Web Rule Language (SWRL), and SPARQL queries.

For example, a SWRL rule in the ontology captures the business logic that the creationDate for any media used to document a project must come after the startDate of the project:

```
startDate(?p, ?psdt) ^ hasDocumentation(?p, ?doc) ^
creationDate(?doc, ?crdt) ^ swrlb:lessThan(?crdt, ?psdt) ->
hasStatus(?p, RequiresAudit)
```

To test this logic, we created some test data (unrelated to any actual NGO or project) for a food distribution project in Pradesh. The documentation for this test project includes a video that was created before the start of the project. This causes the SWRL rule to fire and the RequiresAudit to be added to the project status. One benefit of using SWRL is that explanations can automatically be generated for inferences. Figure 9 shows the explanation generated in Protégé when clicking on the RequiresAudit status for this project.

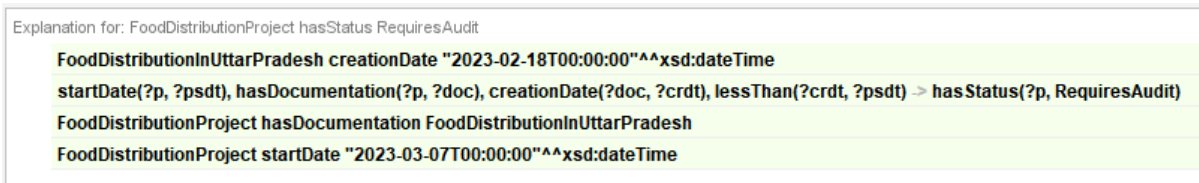


Figure 9: Explanation generated for project status: RequiresAudit

3. Development Process

The project to date has been completely driven by volunteers. Work on the NGO module began in September of 2022. Work on the knowledge graph module began in February of 2023 although the UN SDG ontology was developed independently in 2022. [16] The project has followed an agile development process. The developers of the KG module consisted of one experienced ontology developer working approximately half time and two Berkeley Data Science interns with no previous experience in semantic technology working a few hours a week as permitted by their course load. In terms of Full Time Equivalents (FTEs) this is less than one FTE from February-June 2023. The initial ontology was developed in the Protégé ontology editor. The SDG ontology was imported using the standard Protégé ontology import feature. The data on NGOs was initially imported into the ontology using the Cellfie Protégé plugin. [15]

The initial SPARQL queries to connect SDG Goals, Targets, and Indicators were done with the Snap SPARQL Protégé plugin. However, the SPARQL implementations in Protégé are primarily intended as an introduction to SPARQL, not for project work. In addition, Protégé itself is a modeling tool not a database. Hence, to support the large amount of data, more complex SPARQL queries, knowledge graph visualization, and utilization of Python the AllegroGraph graph database product from Franz Inc. was utilized. [20] In June 2023 we changed our upload process to utilize a Python function to read the CSV files exported from the relational database. The headers of the CSV file have the names of the columns which correspond to the appropriate data properties described above. Thus, we can utilize the same Python function for all the different CSV files.

The immediate future milestones for the project are:

1. Integrate data from Wikidata and replace many of the current properties in the Geography ontology with properties from GeoSPARQL.⁵ We initially created properties such as `containedIn` and `contains` to model relations between states/territories and cities/villages/districts before realizing that GeoSPARQL models these relations in a standard way and provides additional capabilities that will be useful in the future.
2. Create a more sophisticated statistical algorithm to match NGOs and CSRs to SDGs. The current algorithm is fairly simple and only matches the highest-level UN SDG goals. We are developing a more sophisticated statistical algorithm that we believe will both be more accurate and will map to targets and indicators as well as the high-level goals.
3. Release an open-source system that integrates data about NGOs and CSRs using the UN Sustainable Development Goals model. This release will include a GUI that allows non-technical users to generate SPARQL queries by filling out a form that describes parameters such as the SDG metrics they want to search for, the size of the NGO, locations, etc. This GUI has already been developed using the QT GUI Python library. We have a small amount of additional work to do to write the appropriate SPARQL query based on the parameters provided by the user.

The planned August 2023 release will be the most comprehensive data source on NGOs in India because it integrates data from all of the largest public NGO databases, provides significant cleanup of the data, and supplements it via SPARQL queries and the UN SDG ontology. The user friendly GUI will provide the formal rigor and reasoning of the ontologies to non-technical users.

4. Discussion

This project illustrates the benefit of an agile, reuse-based development model when using ontologies for real problems. This is a different model than the model used to create repositories of reusable vocabularies which relies on an overarching upper model to ensure consistency across the vocabularies. The DaanMatch KG project to date has had the equivalent of less than 6 person months

⁵ <https://www.ogc.org/standard/geosparql/>

of development but has created a system that demonstrates the goals of the DaanMatch project to utilize the UN sustainable development goals to match NGOs with CSRs and to replace traditional documentation with media and metadata recorded in real time as part of the execution of an NGO project. It does this with real world data collected by the DaanMatch team as well as test data generated by the KG team. The relational data model of the NGO module and the ontology model of the CSR module have been aligned via data properties as described in section 2.1 and the integration of the most recent data collected and generated by the NGO module into the CSR module is complete.

The Agile approach to ontology development (also demonstrated in [10]) utilizes SPARQL to automate many tasks traditionally done by hand. Rather than focus on a top-down methodology that assumes that the “correct” model can be developed before any data has been imported, the Agile approach recognizes that good design is both a bottom up and top down process. [8] I.e., the structure of data from systems that will populate the knowledge graph often impacts the design of the ontology as much as the domain and reusable vocabularies.

In addition, we are currently working with a data science graduate student from UC Berkeley who is accessing various possible uses of machine learning to further help realize the DaanMatch vision. The first result of this collaboration was a more sophisticated statistical algorithm for matching organizations to SDGs. Possible future opportunities include:

- Utilize deep learning models such as ChatGPT to automate the creation of various forms required for NGOs to receive funding grants. As described above, many of the smaller NGOs, while providing excellent services, lack the business skills to negotiate the existing bureaucratic process.
- Utilize ML to analyze the content of Prov:Entities (e.g., video and audio files) to determine if they may require human auditing due to potential fraud or error in documentation. One of the biggest issues that we anticipate with the long term DaanMatch vision is *change management*. People don't like change. They especially don't like change when it involves bureaucratic procedures. The case for corporations to adopt the DaanMatch model will be significantly aided if it can be demonstrated that the new approach not only makes life easier for NGOs so that they can focus more of their time providing services while also providing equal or superior levels of auditability with less effort via the OWL reasoner, SPARQL engine, and Machine Learning.

The project demonstrates how to use semantic technology for the benefit of society to achieve a high level of productivity, flexibility, and integration with traditional systems and data.

5. Acknowledgements

This work was conducted using the Protégé resource, which is supported by grant GM10331601 from the National Institute of General Medical Sciences of the United States National Institutes of Health. Thanks to Franz Inc. (<http://www.allegrograph.com>) and its help with AllegroGraph and Gruff.

6. References

- [1] Credit Suisse Research Institute, "Global Wealth Report," 2023. [Online]. Available: <https://www.credit-suisse.com/about-us/en/reports-research/global-wealth-report.html>. [Accessed 26 June 2023].
- [2] Centre For Policy Research, "The Evolution of India's Welfare System from 2008-2023: A Lookback," 2023. [Online]. Available: <https://accountabilityindia.in/publication/special-edition-2023-accountability-initiative-centre-for-policy-research/>. [Accessed 26 June 2023].

- [3] Development Monitoring and Evaluation Office, Government of India, "Social Impact Assessment of Corporate Social Responsibility in India," April 2021. [Online]. Available: https://dmeo.gov.in/sites/default/files/2021-11/Report_on_Social_Impact_Assessment_of_Corporate.pdf. [Accessed 26 June 2023].
- [4] N. R. Kumar, Interviewee, *Zoom Interview with AVP - Corporate Communications & CSR, Indian Institute of Corporate Affairs with Cara Aralleno on issues facing Indian CSR programs*. [Interview]. 26 March 2021.
- [5] H. Baykara, "Funding the Frontlines: The Value of Supporting Grassroots Organizing," 25 August 2016. [Online]. Available: <https://philanthropynewsdigest.org/features/commentary-and-opinion/funding-the-frontlines-the-value-of-supporting-grassroots-organizing>. [Accessed 26 June 2023].
- [6] D. Tripathi, Interviewee, *CEO Assam State Disaster Management Authority*. [Interview].
- [7] S. Gyanendra, D. Tripathi and D. M. Ranjan, Interviewees, *Discussions with experts on funding of NGOs in India, DaanMatch project at UC Berkeley Skydeck*. [Interview]. September-November 2019.
- [8] K. Beck, *Extreme Programming Explained*, Boston, MA, USA: Addison-Wesley, 2004.
- [9] C. Mullins, "Extract, Load, Transform (ELT)," TechTarget.com, January 2020. [Online]. Available: <https://www.techtarget.com/searchdatamanagement/definition/Extract-Load-Transform-ELT>. [Accessed 27 April 2023].
- [10] M. DeBellis and B. Dutta, "From ontology to knowledge graph with agile methods:: the case of COVID-19 CODO knowledge graph," *International Journal of Web Information Systems*, vol. 18, no. 5/6, 5 October 2022.
- [11] A. Singhal, "Introducing the Knowledge Graph: things, not strings," Google, 16 May 2012. [Online]. Available: <https://www.blog.google/products/search/introducing-knowledge-graph-things-not/>. [Accessed 8 May 2023].
- [12] Prov W3C Working Group, "PROV-Overview," W3C, April 2013. [Online]. Available: <https://www.w3.org/TR/2013/NOTE-prov-overview-20130430/>. [Accessed 8 May 2023].
- [13] UN Dept. of Economic and Social Affairs, "The 17 Goals," 2022. [Online]. Available: <https://sdgs.un.org/goals>. [Accessed 27 April 2023].
- [14] United Nations, "Open SDG Data Hub," 2022. [Online]. Available: <https://unstats-undesa.opendata.arcgis.com/>. [Accessed 27 April 2023].
- [15] M. O'Connor, C. Halaschek-Wiener and M. A. Musen, "Mapping master: a flexible approach for mapping spreadsheets to OWL," in *9th International Semantic Web Conference (ISWC)*, Shanghai, China, 2010.
- [16] M. DeBellis, "UN Sustainable Development Goals Ontology," 2022. [Online]. Available: <https://www.michaeldebellis.com/post/unsdg-ontology>. [Accessed 27 April 2023].
- [17] A. Blumauer and H. Nagy, *The Knowledge Graph Cookbook: Recipes That Work*, Vienna, Austria: Monochrom, 2020.
- [18] Franz Inc., "AllegroGraph Freetext Indexing," 22 March 2023. [Online]. Available: <https://franz.com/agraph/support/documentation/current/text-index.html>. [Accessed 27 April 2023].
- [19] J. Aasman and K. Cheatham, "RDF browser for data discovery and visual query building," in *Workshop on Visual Interfaces to the Social and Semantic Web (VISSW2011)*, Palo Alto, CA, USA, 2011.
- [20] Franz Inc., "Allegro Graph Graph Database," [Online]. Available: <https://allegrograph.com/>. [Accessed 27 April 2023].