Hezo: An Intelligent Social System for Extreme Scale Organizational Collaborations

THESIS

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science
in the Graduate School of The Ohio State University

By
Ramya Jadda, B.Tech.
Graduate Program in Computer Science and Engineering

The Ohio State University
2011

Thesis Committee:
Jay Ramanathan, Thesis Advisor
Rajiv Ramnath, Academic Advisor
Abstract

How can entrepreneurs or community members discover across un-linked populations and organizations and form a community focus, access knowledge and complete value-adding interactions? This thesis proposes an Intelligent Enterprise 2.0 system called “Hezo” that leverages Technology-Mediated Social Participation through Social Networking technologies to facilitate collaborations that add value. “Hezo” moves beyond existing Social Searching and Collaboration Tools by modeling and semi-automating the cognitive process of discovering goals and resources to enable systematic completion of Extreme Scale Collaborations (ExSO) that add value. We present an ExSO Framework with an example scenario that demonstrates ExSO collaboration advantages. While existing systems support resource discovery by social searching techniques or taxonomy-based Techniques, Hezo connects un-linked populations based on “goal-based” and “role-based” recommendations. We have used Semantic Technologies (RDF and SPARQL) to demonstrate the “goal-based” matching engine.
Dedication

To Dad, Mom and Akai
Acknowledgements

I sincerely thank my professor’s, Dr.Jay Ramanathan and Dr.Rajiv Ramnath for guiding me throughout this research. I also thank my friends in CETI (Center for Enterprise Transformation and Research) group for all their help and cooperation.
Vita

June 2008………………………………………………..Bachelor of Technology,
JNTU Hyderabad, India

June 2010-present……………………………………..Graduate Research Assistant
C.E.T.I,
The Ohio State University

Fields of Study

Major Field: Computer Science and Engineering
# Table of Contents

Abstract ......................................................................................................................... ii  
Dedication .................................................................................................................... iii  
Acknowledgements ...................................................................................................... iv  
Vita ................................................................................................................................ v  
Chapter 1: Introduction ................................................................................................. 1  
  1.1 Extreme Scale Organizational Collaborations ................................................ 1  
  1.2 Our Approach ................................................................................................. 2  
  1.3 Contributions .................................................................................................. 3  
Chapter 2: Problem Statement ....................................................................................... 5  
  2.1 Technical Problem Statement ........................................................................... 6  
Chapter 3: Motivating Project and Problem Analysis ................................................... 8  
  3.1 Motivating Project: MyNeighborhoods ............................................................... 8  
  3.2 Requirement Insights from MyNeighborhoods ................................................. 10  
  3.3 Social Mediation and Non-Routine Service Evolution ...................................... 12  
  3.4 Unique Requirements of Extreme Scale Organizational Collaborations ....... 13  
Chapter 4: ExSO Framework ...................................................................................... 16  
  4.1 ExSO Framework Requirements ....................................................................... 17  
  4.2 ExSO Flowchart ................................................................................................ 19  
Chapter 5: Hezo Architecture and Domain Ontology ................................................. 24  
  5.1 Hezo Knowledge ............................................................................................... 24  
  5.2 Hezo Roles ......................................................................................................... 25  
  5.3 Hezo Ontology ................................................................................................... 28  
  5.4 Hezo Design Architecture ................................................................................. 33
List of Figures

Figure 1: Introduction.................................................................................................... 7
Figure 2: MyNeighborhoods UI.................................................................................... 9
Figure 3: ExSO Framework .......................................................................................... 16
Figure 4: ExSO Flowchart........................................................................................... 19
Figure 5: Hezo Roles................................................................................................... 26
Figure 6: Hezo Ontology 1.......................................................................................... 28
Figure 7: Hezo Ontology 2.......................................................................................... 31
Figure 8: Hezo Architecture........................................................................................ 33
Figure 9: Crawler Agent.............................................................................................. 34
Figure 10: Knowledge Module ................................................................................... 36
Figure 11: Social Matching Agent .............................................................................. 38
Figure 12: Goal Matching Agent ................................................................................ 39
Figure 13: Interaction Module..................................................................................... 40
Figure 14: Messaging Agent....................................................................................... 41
Figure 15: User Interface – Bulletin Board ................................................................. 51
Figure 16: User Interface – Interactions...................................................................... 52
Figure 17: User Interface – Event ............................................................................... 53
List of Tables

Table 1: ExSO Scenario................................................................. 14
Table 2: Interaction 1 ................................................................. 66
Table 3: Interaction 2 ................................................................. 67
Table 4: Framework Support.................................................... 68
Chapter 1: Introduction

How can an entrepreneur or a community member discover un-linked populations and organizations to form a community of focus, to access knowledge and to complete value-adding interactions? This motivating scenario is one of many which Columbus area public, academic, business organizations and individuals encounter while trying to collaborate on community-based problem solving. This thesis focuses on designing a social system that tries to leverage Social Searching and Technology-Mediated-Social Participation across and within the extended enterprise and communities to facilitate "Role-based" Extreme Scale Organizational Collaborations (ExSO). We propose a system called “Hezo”, which moves beyond existing Social Searching and Collaboration Tools by modeling and semi-automating the cognitive process of discovering goals and resources to enable systematic completion of ExSO Collaborations for community development.

1.1 Extreme Scale Organizational Collaborations

As with any bustling metropolitan area, there are many Columbus area public, academic, business organizations and individuals that are collaborating to solve community-based problems. Often they act in the same target area with little
knowledge of each other. There are thus collaborations, which we call “Extreme Scale Organizational Collaborations (ExSO)” where entrepreneurs or community members work in un-linked silos. These ExSO Collaborations often have the need to discover individuals and organizations that have the intent to collaborate and also discover how to achieve value as the collaboration proceeds. Thus the goal of “Hezo” is to dynamically network across organizations to find potential collaborators and coordinate actions for successful completions. By doing so, the eventual outcome is to find potential collaborators and coordinate actions for successful completions.

1.2 Our Approach

In this thesis, we have used the three following approaches.

- **“Goal-Based” Modeling:** Over the past decade, goal models have been used in Software Engineering to capture early software requirements, business objectives and design qualities[1]. Since, collaborations are started to achieve some requirement or objective, we use “goal-based” modeling to capture these requirements.

- **Technology-Mediated Social-Participation:** “Technology-mediated social-participation (TMSP) systems, such as TopCoder and StackOverflow, allow a vast user base to collaborate to solve difficult problems. They provide unprecedented opportunities, by supporting closer coordination among larger groups of people, to solve important social problems while increasing the collective intelligence of the community” [2]. “To Facebook message or to e-mail? Facebook Message.” At
At least a report from Nielsen Online concluded that people spend more time on social networking sites than their e-mail accounts [1]. The social networking phenomenon is dramatically changing the way people behave and communicate. The phenomenal growth of social networking and user’s willingness to co-create content on the web opened up new opportunities for collaborations. TSMP can be effectively applied to solve Community Problems where resources and cost is often a big constraint.

- **“Role-based” Matching:** Existing resource-matching techniques such as social search engines available on the Internet or within enterprises applications are not sufficient to match resources in ExSO collaborations. They match people only based on “what they know” i.e. either their skill or their interest. We introduce a novel matching technique which finds collaborators based on their level of participation in previous collaborations.

## 1.3 Contributions

In this thesis, we have proposed

- A framework that enables these Extreme Scale Organizational Collaborations to successfully complete and add value
- A system architecture-“Hezo” that supports this framework
- Implemented a “Keyword-based” matching engine on the “C.E.T.I” website to direct customers and students questions to the appropriate resources (playbook chapters).
Chapter 2: Problem Statement

Enterprise 2.0 applications-Business collaboration and social software---have moved from the hype phase into serious consideration and adoption at most companies. While most of them work well when collaborations are well-defined i.e. the goals are clearly identified and may/may not have identified the resources, they fail to support ExSO collaborations where the goals and resources are not explicit. HEZO goal is to enable Extreme Scale Organizational (EXSO) collaborations using the following key features:

1. Discover shared interests between potential beneficiaries and providers who are not linked explicitly, but have the potential and intent to engage in interactions that create value.

2. Facilitate and visualize the co-discovery co-commit co-create co-deliver progress of the interactions and performance.

There are many social networking applications like Aardvark and Gravy that helps in discovering people based on the expertise they can provide [6]. Hezo tries to suggest potential collaborators based not only on affordances of assets or resources held by different users of the system, by searching a network of people but also by their level of participation (what we call roles) in previous similar projects. In other
words, the system makes role-based collaborative recommendations. The first key feature focuses on matching/discovering potential collaborators and the second key feature focuses on creating value from the collaboration. We have identified that collaboration (for achieving a main goal) often comprise of sub-goals (interactions) which when completed result in some useful value. Each such sub-goal often goes through a cycle of co-commitment, co-create and co-deliver. Hezo facilitates and visualizes the co-discovery, co-commit, co-create and co-deliver phases of the interactions and performance.

2.1 Technical Problem Statement

As the use of social networking sites within organizations to improve productivity is more widely accepted, we propose a system which uses the semantic power of Web 2.0/3.0 technologies to help semi-automate extreme-scale collaborations within and across enterprises by providing a Spoken Dialogue Technology based Conversational User Interface to identify clear goals to complete a collaboration and make useful recommendations about resources and project semantics based on similar collaborations and thus building a “web-of-conversations”.

• How can we harness the power of Web 3.0 to successfully meet the requirements of ExSO Collaboration?

• How can we progress towards 3.0 with ontology for Technology-mediated collaboration using Web 3.0 technologies?
• How can we facilitate these conversations even when the individuals are not initially linked into a network?

• Further, we will provide scenario that will illustrate that technology mediated social participation platform will help identify unprecedented opportunities to collaborate across enterprises to solve problems.

Figure 1: Introduction
Chapter 3: Motivating Project and Problem Analysis

[7] Introduces the concept of “Social Media Governance”, According to the author “While the political sciences analyze the interplay between action, interaction and institutional frameworks, the discussion on governance in economics revolves around governance configurations and their costs, efficiency and competitive advantages”. The paper claims that although many organizations claim to have strategies for social media communications, 90% do not have any explicit regulatory frameworks. The next section talks about how the City of Columbus has started bridging this gap with MyNeighborhoods proposal.

3.1 Motivating Project: MyNeighborhoods

The research started with MyNeighborhoods, a GIS based application proposed by the City Of Columbus to [8]:

- Bring government closer to the residents of Columbus by engaging them in civic participation;
- Help them become empowered to make positive contributions to their communities;
- To involve disadvantaged populations in civic discourse, processes, and opportunities.
We started the first phase of work towards achieving these goals by developing a GIS based application that could service “Informational requests” from the citizens. Figure 2 shows a screenshot of the application.

Figure 2: MyNeighborhoods UI

The application provided two kinds of Information 1) Points of Interests in a Neighborhood like parks, basketball courts, schools, hospitals etc. and 2) City Services for a Neighborhood like Civic Associations, Trash Collection days etc. The following figure shows the MyNeighborhoods User Interface. The application needs
very little user interaction with the system as the queries are just informational. The interaction is restricted to 1) selecting the neighborhood based on address or just selecting the neighborhood from a dropdown list, 2) selecting a service layer they want to see on the map (Point of Interests or City Services) and 3) The information they want to see on the map. Clearly, User Interface for “Hezo” should support more interactions between users and also, users and the system.

3.2 Requirement Insights from MyNeighborhoods

A competitive analysis of award winning websites has helped us identify the existence of important requirements and objectives in providing the desired user experience. The eGov websites that are successful according to the Center for Digital Government and popular among citizens are the ones which have organized their services around the citizens rather than their departments. They concentrate on helping citizens with varied interests find what they are looking for by connecting them with the appropriate services or information in the least amount of time. The recognition of these sites is based on goals such as:

Data Federation: Data is often located on different servers and maintained by different departments. This data has to be manually updated in Department of Technology’s server. For users to get the most up-to-date data, this process has to be automated. Also, not all data about the Neighborhood is maintained by the City, for example, parks like National parks, Federal Parks are not maintained by the City and hence they do not maintain their information.
**Data Quality:** Since not all data is maintained by the City, some data is fetched from Google server, which they believe is more updated than the City servers. As Google server returns data based on keyword search, not all data is filtered. For example, when searching for “Fishing places”, we found that Google returns Fishing stores.

**Target different user groups:** This involves identifying the different user groups who would interact with the portal and provide them with the appropriate view of information and services. The goal is to create satisfaction within every user group, that the agency is committed to addressing their problems and needs.

**Good search:** Providing good search capabilities is necessary to get the citizens to find the relevant information with minimal effort, thereby increasing satisfaction.

**Simple and logical user interface:** Even if relevant services and information are present, complex or difficult interfaces will preclude their use. So, the goal is to design simple and logical interface for the purpose of encouraging adoption as well as ongoing use. Having GIS Visualization embedded in the sites attracts more users, as they could locate directions, the location of the resource etc.

**Seamless integration between public and private sectors:** The citizen who visits the portal for accessing a particular service should not be bogged down by the details of whether the service is public or private. For a citizen, the goal should be to achieve a one-stop portal which would address all the issues that the citizen may have in each context, regardless of service location.
3.3 Social Mediation and Non-Routine Service Evolution

Again, based on an analysis of award-winning websites, we discovered that Government websites are not just looking at providing limited information to the citizens, but now attempting to progress to higher-and-higher levels and goals, the services requested by citizens have progressed from informational → transactional → process status → participatory → collaborative planning → collaborative interactions.

- **Informational**: Websites with products and services (MyNeighborhoods)
  
  E.g. Citizen’s queries regarding Museum location and directions.

- **Transactional**: Purchase, scheduling of services.
  
  E.g. Citizen’s buying Museum ticket.

- **Process status**: Service delivery status.
  
  E.g. automatically, update website that the Museum is crowded or under repair.

- **Participatory**: What do you think of our service?
  
  E.g. Citizen can participate and offer opinions on new Museum programs.

- **Collaborative planning**: Can you help design our service?
  
  E.g. Citizens can engage deeply in serious gaming i.e. translation games for exploration of Museum exhibits that allow us to sense behaviors.

- **Collaborative interactions**: Can we identify how we can co-create value?
  
  E.g. Citizen is matched with resources from many different organizations that address his needs.
3.4 Unique Requirements of Extreme Scale Organizational Collaborations

Based on several focus group sessions, requirement planning and strategic planning projects with the City of Columbus, we found that collaborations identifying and meeting new goals and innovation potential often necessitates discovery:

- Potential needs are not explicit; this means we do not know the key words to search on web sites and other linked-node type systems (Aardvark, LinkedIn) etc. Participants in potential collaborations do not know of each other. For example, the fact that an NGO could be an excellent resource for an academic service learning course is not easily evident in the websites. Another example is a budding entrepreneur could more likely achieve success through new applications by working collaboratively and reduce risk. We need to find ways to make tacit knowledge explicit (e.g. use human computation).

- There were also pragmatic issues- a very great reluctance to use one more website! We need a brokering (engine) type approach and related site (analogous to Google™) that will also push information to people. (e.g. via email)

- Based on a detailed analysis of a successful collaboration between the Linden Community, Columbus State Community College, Businesses, and The Ohio State University reflected in Table 1, we found it is useful to distinguish a conversation that goes ‘somewhere’. We call this an Interaction. Most importantly we found users discover new Interaction requirements as conversation proceeds spawning other Interaction threads.
• Another important concept is that external events often start multiple Interactions. For example, hurricane Ike’s aftermath caused many Interactions across many public and private organizations, as first responders collaborate to take actions. These Interactions are related by the event. Further, certain sub-interactions and value might get defined during the course of overall interaction, but for others discovery and negotiation continues. More generally, these discovery characteristics are typical of all non-routine service interactions.

• Example Scenario:

<table>
<thead>
<tr>
<th>Stakeholders, Goals, Constraints</th>
<th>Needs (what)</th>
<th>Profiles of Resources/Assets (who, what, when, how much)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greater Linden Development Corp. (GLDC): Advance Linden community OSU CETI: Provide academic curriculum to meet ABET criteria</td>
<td>Need technology projects Need inexpensive pricing Need computers Need space design Need volunteers 📅: 2 years of discovery for this connection Need sponsored technology-based projects</td>
<td>When: Summer What: Training and mentored technology projects When: Scheduled courses What: Service Learning provides free resources for good student learning experiences</td>
</tr>
</tbody>
</table>

Table 1: ExSO Scenario

Continued….  

14
<table>
<thead>
<tr>
<th>1.1</th>
<th>GLDC: Acquire computers without any cost</th>
<th>Need to fix surplus into working computers</th>
<th>What: Surplus, Extra computers for parts When: Surpluses occur</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Businesses: IT organizations, OSU IT, NW, OSUMC: Surplus old equipment</td>
<td>Need to surplus computers</td>
<td>When: Scheduled courses What: Service Learning provides student resources (free) for learning experiences Why: Repair store too expensive</td>
</tr>
<tr>
<td></td>
<td>FreeGeeks@CSCC – part of service learning</td>
<td>Need to train refurbishing computers</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.2</th>
<th>GLDC: Engineering Project/Training services using computers</th>
<th>Need programming</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TechCorps: NGO mission to provide STEM curriculum</td>
<td>Need curriculum</td>
<td>When: Scheduled courses What: Service Learning provides student resources (free) for learning experiences Why: Repair store too expensive</td>
</tr>
<tr>
<td></td>
<td>Friendship Village: Senior citizen activities</td>
<td>Need tutors (can't travel)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1.1.1</th>
<th>ConnectOne TimeWarner City of Columbus</th>
<th>-Need Warehouse computers -Need Skype Computers must access Internet Need to broadband and/or wireless Need access locations Need GIS map of broadband coverage</th>
<th>Why: Need space to store donated computers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Need to learn about ICT requirements in communities Need transportation</td>
<td>Seeds for change Pair up with others that have cars – free</td>
</tr>
</tbody>
</table>

| 1.2.1 | How will students get to locations safely? What will OSU students learn? | Need to learn about ICT requirements in communities Need transportation | Seeds for change Pair up with others that have cars – free |
Chapter 4: ExSO Framework

Our goal with ExSO proof of concept is to show we can support collaborations that create new possibilities and opportunities which wouldn’t otherwise exist. The ExSO software systems should help discover potential collaborators and also provide a social networking platform that would facilitate the actual collaboration. The following figure shows a ExSO collaboration framework.

Figure 3: ExSO Framework
4.1 ExSO Framework Requirements

High-level Functional Requirements: Based on the ExSO framework, Hezo should support the following requirements.

- **Co-discover:**
  - Suggest potential resources based on user’s skill or asset’s affordances.
  - Role based resource matching. Hezo should suggest potential collaborators who might be interested in participating in the interaction based on their level of participation in similar interactions.

- **Collaborate:**
  Hezo should provide a social networking platform that provides Interaction management i.e. maintaining hierarchy and relationships among interactions and facilitate the execution of tasks.

- **Co-commit:**
  - Provide a more structured “Forum” like space for people to interact to discover the requirements based on the goals.
  - Suggest Project Semantics for based by matching the goals of the interaction with previous interactions in the Knowledge Base.

- **Co-create:**
  - An interaction may have to finish a task to achieve its goal.
  - Provide services for task execution i.e. task management tools.
  - Assign Roles to users based on their level of participation.
• **Co-deliver:**
  
  • Extract project semantics of completed interactions and build a knowledge base with all project semantics and users involved in the interaction along with the roles they were assigned.
  
  • Provide tools to check if the interaction has achieved its goal by, say a voting mechanism.
  
  • Collect feedback from all the participants of that interaction.

• **Knowledge Management:**
  
  • Browse project semantics like, start time, end time, tasks involved, resources etc. for all the interactions.
  
  • Provide Visualization tools that help better decision making.
  
  • Build a knowledge base of roles associated with each interaction.
  
  • Provide services for rules, tools and methods to identify trends and patterns to facilitate collaborations.
4.2 ExSO Flowchart

Figure 4: ExSO Flowchart
Model:

We believe a good model can serve as a solid foundation of a well designed collaboration system. Thus, we propose the following model in the Figure 5, which captures the interactions happening within collaboration.

As shown in the Figure 3, a successful collaboration should go through four phases: Co-Discovery, Co-Commitment, Co-Creation and Co-Delivery. The process of a collaboration proceeds as follows.

**Co-Discovery:**

Based on the goals and needs of organizations/individuals, a good collaboration system should be able to match potential needs with appropriate resources. The process of matching can be initiated when an organization or individual posts a need or goal in the form of an interaction. The system at this point analyses the semantics of the posted need/goal and tries to suggest potential collaborators or resources using the matching agent of Hezo. This matching is done based on the skill set/affordances and resources held by the users of the system. The system also tries to suggest users based on their level of participation in similar projects. If the system fails to suggest any partners we use crowdsourcing and facilitators administration to find potential matches for the project.
Co-commitment:

If the resources are not matched at this stage, the interactions enter this stage to discuss the “how” parts of projects which do not have pre-determined requirements (They only have an explicitly stated goal). But, if the needs and services are properly matched, the collaboration will proceed to this next phase where the collaborators can discuss and agree on requirements. Some of these interactions can initiate new interactions during the discussion of commitment. We call this way of generating a new interaction as dynamic threading. We will talk about dynamic threading in details later in this section. Right now you can consider it as starting another new interaction based on the current one. If people fail to achieve co-commitment, they should check whether the partners or resources are correctly chosen. If not, they should go back to re-matching goals/needs and resources.

Co-Creation:

This phase is the execution part of the collaboration. Based on achieved commitments, actions are taken to meet the requirement. Interactions are of two types: one which are started to discover the “how” part of the projects and others that have a specific task to achieve, which in turn creates value. Only interactions that have specific task to achieve proceed to this stage. Periodically, partners check whether the requirements are satisfied. If not, they can either go back checking whether commitments could possibly be accomplished or go back further to previous phase. A new interaction can be started if everything mentioned above failed.
Co-Completion:

In this phase, the collaborators vote if the interaction has achieved its goal or not. If they vote that the goal is achieved then the interaction is complete. Once, the interaction is completed, we extract project semantics of this interaction and build a knowledge base with all project semantics and users involved in the interaction along with the roles they were assigned. We discuss the knowledge base in detail later section. In this phase, we also collect feedback from all the participants of that interaction about the system as well as other participants.

Knowledge Base:

Hezo keeps profiles of the users and assets. If the user registers with Hezo directly, s/he is asked to fill out a profile, which Hezo uses when it tries to match based on skills. Hezo also extracts user’s skill from their social networking profiles, from their homepages and also from the content they create on Hezo. And, as the interactions are completed, the system updates the knowledge base with any project semantics it might extract from the interaction. It also, updates the different roles different users play per interaction. Hezo uses this Knowledge base to make resource and project semantic suggestions to users.

Dynamic Threading:

Say, you are looking for a partner to write a business plan together. You have the goal of the plan while everything else is vague. You two go through co-commitment and
co-creation. After some discussion, you find the problem statement is not feasible, which means everything should start over. So you close the current interaction and start a new one. And now say, after some discussion or based on the suggestion made by Hezo, you conclude that some other sub-goal has to be met to complete this interaction. Again, we start a new interaction which will be fired by the current interaction. Hezo, keeps track of all the relations between the interactions which when completed will help extracting the project semantics. As we have seen in the example scenario, a successful collaboration may come through a lot threads to proceed. Each thread has its own purpose. So at the end of the project, you will have a thread graph. Each node in the graph is a thread and they are connected to each other. This graph assists you navigate through the lifecycle of the project.
Chapter 5: Hezo Architecture and Domain Ontology

This chapter presents the Hezo system architecture. We first introduce the Hezo Domain Knowledge and the Ontology.

5.1 Hezo Knowledge

In this section, we introduce the core ExSO objects. Below are the 5 core objects that we have identified as part of ExSO collaborations:

- **Event:** Start of any significant collaborative work. An individual/organization can start an event, when they have a goal and are looking for partners to collaborate to achieve this goal. More often than not, the individual/organization tries to add value to their community/organization by achieving this goal.
  
  **E.g.** Developing Linden Community kick off meeting. In the example, scenario we have provided earlier, Representatives from Linden Community can start an event with the goal of developing their community.

- **Interaction:** The different conversation topics people start for an Event. Once, an individual/organization has started an event, users of the system discuss the event by starting interactions. They identify sub-goals that are required to achieve the final goal of the collaboration. As need sub-goals are discovered users start new interactions.
E.g. Going back to the Linden Community Scenario, once the event has been started, people start conversations to discuss how the goal can be achieved. “Need to set up a computer center” is a sub-goal that is discovered in one of the previous interactions.

- **User:** Any registered/anonymous user visiting the website or Organizations that would wish to contribute. Registered users could again be users who are directly registered with the system or users who use popular social networking authentication to use the system. There will be a profile associated with each user.

  E.g. From the scenario presented earlier, some of the users are CETI, Linden Community, OSU Medical Center, etc.

- **Assets:** Any tangible resources a USER may have. These assets may be assigned different roles based on the need they are fulfilling.

  E.g. Computers for Donation, Extra Space

### 5.2 Hezo Roles

ExSO collaborations usually require a team of experts that must work together on a solution [20]. We have defined the notion of roles for users who act as experts and come together to complete an interaction.

**Roles:** Rule based assignment to Users based on their contribution to previous or current projects and also based on the trust they have earned. Users can play multiple roles across different interactions.
The figure below (Figure 4) shows the main roles that a user can be assigned to.

![Figure 5: Hezo Roles]

- **Sponsoring organization**: establishes policy, authorizes (e.g. Columbus school district)
- **Beneficiary**: directly gets value from the project (e.g. adult, community)
- **Participant**: contributes service and learns (e.g. student or adult)
- **Coordinator**: approves scope, resources and helps ensure value is created (e.g. faculty)
- **Donor**: Provides assets that can include time, equipment, finances (e.g. company xyz)
- **Friend**: observes and helps, acts as an ambassador (e.g. professional volunteer)
- **Facilitator**: triages and administrates cases that need attention
- **Volunteer**: contributes services of his/her own free will (e.g. student, faculty, adult)

**Figure 5: Hezo Roles**

**Sponsoring Organization**: Any organization that establishes policies and authorizes programs.

E.g. From our example scenario, Columbus School District has to authorize the course curriculum.

**Beneficiary**: An individual/organization that directly gets value from the project. Often, beneficiary is the one who starts an event. Beneficiary comes up with a goal and initiates the collaboration process.

E.g. Again from our example scenario, Linden Community is the beneficiary.
**Participant:** An individual who contributes his services and learns. Participants are individuals who contribute their services in return for some personal gain/value.

E.g. Again from our example scenario, the people of Linden community participate in the Service Learning Courses, to improve their skill.

**Coordinator:** Approves scope, resources and helps ensure value is created. Coordinators are people who manage an interaction to ensure the goal of that interaction is achieved. Any interested participant willing to take the responsibility of an interaction is a Coordinator.

E.g. Again from our example scenario, CETI coordinated the “Setting up a Computer Center” Interaction.

**Donor:** Provides assets/resources that can include time, equipment, finances. Donor is an individual or organization who donates assets/resources.

E.g. Again from our example scenario, OSU Medical Center is a donor.

**Friend:** Observes and helps, acts as an ambassador. Friend is any individual who interacts with the site. The first interactions are usually in the role of a friend.

E.g. Again from our example scenario, anyone who replies to the interaction posted will be a friend.

**Facilitator:** Triages and Administers cases that need attention. Facilitators are set of people who have earned the trust of the community and act as administrators. When Hezo fails to match resources, which cause interactions to halt, a Facilitator triages and administers the interaction.
E.g. CETI could be a Facilitator for our example scenario.

**Volunteer:** Contributes services of his/her own will. Volunteer will be someone who has earned some trust of the community and whose services greatly help the collaboration.

E.g. An OSU faculty, who builds a curriculum for a Service Learning Course.

### 5.3 Hezo Ontology

In this section, we introduce the Hezo Ontology. Figure 6 shows how the core objects interact with each other using a UML static diagram.

![Figure 6: Hezo Ontology 1](image)

UML structure diagrams show the static structure of the system being modeled, focusing on the elements of a system, irrespective of time. Static structure is
conveyed by showing the types and their instances in the system. Besides showing system types and their instances, structure diagrams also show at least some of the relationships among and between these elements and potentially even show their internal structure [14].

The diagram shows how each object is related to other objects. Most of these relationships are dynamically determined by a rules engine. As the requirements are discovered dynamically as the interaction proceeds, the relationships among the objects are also dynamically determined based on a set of rules that work around the interactions. The rest of the section explains the relationships between the objects and how these relationships are dynamically determined.

As mentioned before, Event is the start of any collaborative work. Every Event has a goal that is explicitly stated and the goal of the collaboration is to achieve that goal. And this goal is usually achieved by achieving multiple sub-goals. The users can start multiple interactions concurrently, to achieve these sub-goals. Hence, every event consists of one or more interactions.

Each interaction, based on the sub-goal it is trying to achieve, might spawn other interactions, i.e. new interactions are started when a sub-goal is not achieved or when next sub-goal is started on achieving this. (Next section talks more about Interactions).

Users are any organizations/ people that use the system. Each user will could interact with the system as an anonymous user, without creating a profile or by
signing in with the system. When users register with the system, they are asked to provide certain information on their skills, resources and interests, which will be used for matching them for potential collaborations.

Assets are tangible resources owned by users in the system. Based on the need these assets could play different roles. Assets are associated with affordances to keep track of the multiple uses of a particular resource.

Roles are the levels of participation of users in any interactions, i.e., based on a users participation he is assigned one of the roles discussed above. So, for every interaction a user who participates in an interaction is assigned a role. And, a user can play multiple roles across interactions. The roles can be divided broadly into Customer and Provider roles for each interaction. Beneficiaries and Sponsors usually act as customers requesting some need. Donors, Facilitators and Coordinator usually act as providers of the request. Here, Friend can act both as a Customer or Provider.

Each User/Asset might play multiple roles across interactions. And, each interaction has multiple roles which in turn are played by different users/assets. Human Computation/Rules determine how to assign these roles to users/assets for each interaction.
Figure 7: Hezo Ontology 2
This figure explains in detail the activity diagram of an interaction. We have found from our example that interactions can be of 3 types.

- Clarification
- Task
- Review

Interactions that are started to discuss how a sub-goal might be achieved are called Clarification interactions. Here, only “how” part of a project/sub-goal is discovered. In a way, project semantics are inferred from these kinds of interactions. They are just “Refine” events. They continue till a solution is found. They do not have inter-dependencies.

The other kind of interactions is Tasks. They are usually started to achieve some kind of a task. Again, some tasks are independent of all other interactions—“Fire Events”, while others are dependent on some other interactions to complete, “Fire and Wait Events”.

32
5.4 Hezo Design Architecture

This chapter describes the proposed Hezo Architecture along with an in detail explanation about each Module and Agent.

Figure 8: Hezo Architecture
5.4.1 User Profiles

A comprehensive knowledge about users is important to match them based on skills. Building these user profiles depends on designing interfaces and algorithms that can collect and learn an extended skills-list of each user. Apart from collecting basic information like their primary skills and interests, we propose a Crawler Agent which can populate the User Profiles.

**Crawler Agent:**

![Figure 9: Crawler Agent](image)
The Crawler Agent does the following [21]:

1) Parse out information from user’s existing online profiles (Other Social Networking sites/ home pages of individuals, blogs, Websites of organizations, Twitter Messages etc.)

2) Content people post on Hezo, i.e. the interactions they contribute to.

5.4.2 Knowledge Module

Knowledge sharing among people is important for efficient online collaborations [22]. Goals have been used in software engineering to model early requirements and non-functional requirements. Goal Analysis consists of decomposing goals into sub-goals though an AND- or OR-decomposition [23]. As these goals and sub-goals are achieved we complete the interactions which were started to achieve these goals and sub-goals. Hezo extracts out project semantics from these completed interactions and their sub-interactions. The goal/sub-goal will be the requirement and the project semantics tell us how the requirement has been fulfilled. As mentioned earlier, we need group of experts (users playing different roles to successfully complete an interaction.
Roles Agent and Project Agent:

Roles Agent applies rules to assign users to different roles. After the interaction is completed it updates the Knowledge Base with all the roles and associated users for an interaction.

The Project Agent extracts the project semantics from a completed interaction and its sub-interactions, to capture the Requirement and sub-requirements of that goal.
and the resources used, the project plan etc and update’s the knowledge base for every interaction.

5.4.3 Matching Module

Social matching systems bring people together in both physical and online spaces. They have the potential to increase social interaction and foster collaboration. Social Matching is the process of recommending an individual to another individual based on some criteria [18]. Many Social networking sites enable individuals to maintain existing social ties and develop new ones with people who share similar interests (Facebook, LinkedIn). As users join online communities and contribute content (Stackoverflow, Wikis), new ways of forming and maintaining relationships are becoming possible [20]. In existing social matching systems, users have a clear knowledge of the goals. But, in ExSO collaborations goals are discovered as the interactions are executed. Hence, suggestions should be dynamically made as the goals are being discovered. Hezo makes 3 suggestions for an Interaction:

1) Skill-based Suggestions

2) Role-based Suggestions

3) Project Semantic Suggestions
Social Matching Agent:

Social Matching agent suggests users based on the keywords extracted by the Dialog Extraction Agent. The Social Matching Agent uses the keyword to search all the profiles and also profiles in the users extended friend network (if the system has their social networking profiles).

Figure 11: Social Matching Agent
Goal Matching Agent:

The Goal Matching Agent suggests project semantics as well as users based on the role they have played in previous projects. The Goal Matching Agent takes keywords from the Dialog Extraction Agent to search the Knowledge Base built by the Knowledge module.

5.4.4 Interaction Module:

Interactions spawn new/sub Interactions as goals and sub-goals are discovered. Interaction module has 2 Agents to manage the interactions.
1) Interaction Agent

2) Dialog Extraction Agent

1) It manages the hierarchy of all the interactions and provides services to maintain/update/check the status of tasks within the interaction.

2) It extracts semantics/keywords from the interaction, which are used by the Matching Agents to suggest resources.
5.4.5 Messaging Agent:

Since, users show reluctance to use one more system or website. Messaging Agent sends emails to the users suggested by the Matching module, requesting them to participate in the Interaction. Whenever new goals are discovered and new suggestions are made based on these goals, the Messaging Agent alerts users that there might be a new Interaction in which they might be interested in participating.
Chapter 6: Related Tools and Methods

In this section, we discuss about tools and methods related to our system.

6.1 Semantic Web Technologies

Semantic Web is a term coined by the World Wide Web Consortium (W3C) director Tim Berners-Lee. It describes methods and technologies to allow machines to understanding the meaning- or ‘semantics’ –of information on World Wide Web[11]. While the term “Semantic Web” is not formally defined, it is mainly used to describe the model and technologies proposed by the W3C. These technologies include the Resource Description Framework (RDF) [12], a variety of data interchange formats (e.g. RDF/XML, N3, N-Triples [13]), and notations such as RDF Schema (RDFS) and the Web Ontology Language (OWL), all of which are intended to provide a formal description of concepts, terms, relationships within a given knowledge domain.

6.1.1 Resource Description Framework

The Resource Description Framework (RDF) is a family of W3C specifications originally designed as a metadata description model. It has come to be used as a general method for conceptual description or modeling of information that
is implemented in web resources using a variety of syntax formats. The RDF data model is similar to classic conceptual modeling approaches such as Entity-Relationship or Class diagrams, as is based upon the idea of making statements about the resources (in particular Web resources) in the form of subject-predicate-object expressions. These expressions are known as triples in RDF terminology. The subject denotes the traits of aspects of the resources and expresses a relationship between the subject and the object. For example, one way to represent the notion of “The sky has the color blue” in RDF is as the triple: a subject denoting “the sky”, a predicate denoting “has the color”, and an object denoting “blue”. RDF is an abstract model with several serialization formats (i.e., file formats), and so the particular way in which a resource or triple is encoded varies from format to format. RDF’s simple data model and ability to model disparate, abstract concepts has also led to its increasing use in knowledge management applications unrelated to Semantic Web activity.

A collection of RDF statements intrinsically represents a labeled, directed multi-graph. As such, an RDF-based data model is more naturally suited to certain kinds of knowledge representation than the relational model and other ontological models. However, in practice, RDF data is often persisted in relational database or native representations also called Triple Stores, or Quad stores if context (i.e. the named graph) is also persisted for each RDF triple. Additional ontology languages can also be built upon RDF.
Ontology is similar to a dictionary or glossary, but with greater detail and structure that enables computers to process its content [15]. More formally, Ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class members). The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application [16]. In the context of knowledge sharing, ontology is a specification of a conceptualization. That is, ontology is a description of the concepts and relationships that can exist for an agent or a community of agents. We design ontologies so we can share knowledge with and among these agents. Key Applications of Ontologies include:

1) Used to specify standard conceptual vocabularies in which to exchange data among systems.

2) Provide services for answering queries.

3) Publish reusable knowledge bases.

4) Offer services to facilitate interoperability across multiple heterogeneous systems.

**FOAF Ontology:**

FOAF is a project devoted to linking people and information using the Web. Regardless of whether information is in people’s heads, in physical or digital documents, or in the form of factual data, it can be linked. FOAF integrates three
kinds of network: *social networks* of human collaboration, friendship and association; *representational networks* that describe a simplified view of a cartoon universe in factual terms, and *information networks* that use Web-based linking to share independently published descriptions of this inter-connected world. FOAF does not compete with socially-oriented Web sites; rather it provides an approach in which different sites can tell different parts of the larger story, and by which users can retain some control over their information in a non-proprietary format [17].

### 6.2 CrowdSourcing:

Coined by Jeff Howe and Mark Robinson, the term crowdsourcing describes a new web-based business model that harnesses the creative solutions of a distributed network of individuals through what amounts to an open call for proposals [9]. It has a potential to exploit crowd of innovators. Even corporate research and development (R&D) for scientific problems is taking place in crowdsourcing way by enabling scientists to receive professional recognition and financial awards [10]. It tries to leverage the mass collaboration enabled by Web 2.0/Web 3.0 technologies to achieve business goals.

- Perceived benefits of crowdsourcing include the following:
  - Problems can be explored at comparatively little cost, and often quickly.
  - The organization can tap a wider range of talent that might be present in its own organization.
• By listening to the crowd, organizations gain first-hand insight on their customer’s desires.

• The community may feel a brand-building kinship with crowdsourcing organization, which is the result of an earned sense of ownership through contribution and collaboration.

Hezo not only uses general Crowdsourcing to get ideas and volunteers for projects from the “crowd” but also based on the suggestions made by the engine; Hezo sends emails to identified users. Here, Hezo tries to focus on small group of experts based on their level of interaction in previous projects. If a new project/need comes in, Hezo crowdsources the goal to the facilitators.

6.3 Human Computation

Human-based computation is a computer science technique in which a computational process performs its function by outsourcing certain steps to humans. This approach uses differences in abilities and alternative costs between humans and computer agents to achieve symbiotic human-computer interaction. The system asks a person or a large group of people to solve a problem, then collects, interprets, and integrates their solutions. In different human-based computation projects people are motivated by one or more of the following.

• Desire to diversify their activity (e.g. “people aren’t asked in their daily lives to be creative”)

• Volunteerism, desire to support a cause of the project
• Desire to be entertained with the competitive or cooperative spirit of a game
• Desire to communicate and share knowledge
• Increasing online reputation/recognition. Builds trust in the online community.

6.4 Content Management System

**Content Management System (CMS)**: A content management system is designed to simplify the publication of web content to web sites and mobile devices — in particular, allowing content creators to create, submit and manage contents without requiring technical knowledge of any Web Programming Languages or Markup Languages such as HTML or the uploading of files. In other words, CMS allows anyone to easily publish, manage and organize a wide variety of content on a website [23]. The main uses of CMS are:

• Allow a large number of people to contribute to and share stored data
• Control access to data, based on user groups (anonymous vs. registered vs. site manager).
Chapter 7: “Keyword-Based” Matching Engine: Implementation

This chapter explains the details of a “Keyword-Based” Matching Engine that has been implemented along with the screenshots of the UI. I used Drupal Content Management System to demonstrate the Matching Function- Match users based on their Assets. I used Jena Semantic Framework to run SPARQL queries on the RDF/XML of content collected from users using the Drupal CMS. The next section briefly discusses the technologies used.

7.1 Technologies Used

• Drupal vs. Plone:

The current C.E.T.I website uses Plone Content Management System, which is based on Python and runs on Zope Application Server, while Drupal is based on PHP /MySQL and runs on Apache/IIS. Configuring the Zope Application Server and developing applications on Plone has a very high learning curve. In contrast, Drupal is easy to understand and Users can do a great deal with minimum training. Also, Drupal 7 has Semantic Web Technologies built in-any content that users create in Drupal can be embedded as RDFa into a Drupal Page.
• RDF/SPARQL:

As mentioned earlier, RDF is used as a general method for conceptual description or modeling of information that is implemented in web resources using a variety of syntax formats. This has two advantages- this RDF can be used by external websites to link this data automatically on their websites. “Many websites like Facebook and Google have already started using RDF to describe their content.” Second, this RDF can be sent used with an external reasoner. SPQARQL provides easy SQL like querying functionality on the RDF data.

• Jena Semantic Framework:

Since, there was no inbuilt SPARQL support in Drupal; I have used an external reasoner, Jena, to run the SPARQL queries. In the future, when more powerful Matching and Rules Engines will be developed Jena will provide reasoner, which will support inferencing based on rules.

7.2 Introduction to Drupal

Drupal is a developed in PHP and uses MySQL as its backend database. I used WampServer 2.1 as the web server that hosts my Drupal website. WampServer is a Windows web development environment. It allows you to create web applications with Apache, PHP and the MySQL database. It also comes with PHPMyAdmin to easily manage the databases.
Drupal Basics:

All content on a Drupal website is stored and treated as "nodes." A node is any posting, such as a page, poll, article, forum topic, or blog entry. Comments are not stored as nodes but are always tied to one. Treating all content as nodes allows the flexibility of creating new types of content such as forms. Drupal has custom modules which can be added on top of the core installation. These custom modules give extra functionality to the Drupal Site, for example, creating new content type using CCK module). The core installation comes with modules for Search, maintaining System Log, User registration and login etc.

Drupal Modules Used:

**CCK Module:** The Content Construction Kit Module allows you to add custom fields to nodes using a web browser. This module is ideal for creating custom forms for User Profiles, Starting new Interactions by collecting needs etc.

**RDF Module:** RDF module provides APIs for developers to manipulate RDF data, as well as output Drupal’s data as RDF/XML. RDF UI allows site administrators to manage the RDF mappings of their site by specifying mappings for the new content types and fields we create using the CCK module.
7.3 User Interface

This section presents the User Interface developed so far. Below are the screenshots.

Figure 15: User Interface – Bulletin Board

This is the first screen that a user sees when they visit the website. Here, the Interactions are grouped under Events. User can browse through the Events and all the Interactions under each Event. If a user wants to participate in any Interaction, he clicks and goes to that Interaction page, where he can post his comments.
Figure 16: User Interface – Interactions

When the user selects one of the Interactions, he goes to the Interaction Page. This page contains the Event under which this Interaction was created and also the description provided when some user created the Interaction. He can then participate by commenting on the Interaction.
If a user wants to start a new Event he can select the Event tab. This will take him to “Create Events” page, where he can enter the title of the Event and an optional description about the Event.
If users want to start a new interaction, they can select the Interaction tab. It will take them to a “Create Post A Question” page. Here, the user will have to select the Event under which they want to create the Interaction, post their question and an optional Description.
The “User” tab allows users to add a new “User” to the system. The User has three fields First Name, Last Name and Assets. We use Assets to match the users.
Figure 20: User Interface - Search

A user can search for other users in the system with a particular “Asset”, by using the Match Tab. Here, the user enters the “Keyword” to search for users with that Asset.

7.4 RDF and SPARQL in Drupal

Adding RDF Mappings to Content Fields:

The User Content Type has Title, First Name, Last Name and Interest as its fields. We enable each field in this content type with RDF Mappings. Foaf ontology is used to describe the fields.
Figure 21: Create User

**FIRST NAME**

**RDF Predicates**

```
foaf:firstName
```

Enter the predicate mappings for First Name here. Predicates should be listed in CURIE syntax, with the vocabulary namespace and predicate. For example, foaf:firstName.

**Datatype**

```
xs:string
```

If there is a datatype for First Name, enter it here. You can use XML Schema datatypes or other datatypes.

**Attribute Type**

```
property
```

If a field contains a literal, such as a string or an integer, then the property attribute should be used. If it contains a URI or node reference, the ref or rev attribute should be used.

Figure 22: foaf - firstName
Now when a user creates his profile, it is stored as node in Drupal. The RDF module generates the RDF/XML. Internally, RDF module uses ARC2 parser which converts the HTML into RDF/XML. This can be accessed at http://localhost/drupal/node/*/rdf.

```xml
<rdf:RDF>
  <rdf:Description rdf:about="http://localhost/drupal/rdf/node/1"/>
    <foaf:firstName rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Ranjan</foaf:firstName>
    <foaf:surname rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Jadda</foaf:surname>
  </rdf:Description>
</rdf:RDF>
```

Here, we see some additional mappings as well; these are created by the core RDF module.
Running SPARQL Queries:

Jena Semantic Framework: Since, there is no reasoner available for PHP; I have used an external reasoner Jena which is based on Java. This is a demonstration of using an external reasoner with Drupal. We use Jena only to query the RDF based on the search keyword. The java code builds a model graph of the nodes of User content type. This graph is queried for the keyword using SPARQL. For future increments, Jena could be a powerful tool to build the rules engine.
Chapter 8: Validation

We validate this thesis in two parts, first we show that current scaffolding supports “Keyword Based Searching” using Semantic Technologies and a Content Management System. We then describe the C.E.T.I implementation of the current scaffolding. Finally, we the validate system design by demonstrating how the proposed Hezo System Design supports the example scenario from Table 1.

8.1 Current Hezo Scaffolding

The current scaffolding provides a simple User Interface for the users to interact with each other. The users can browse all the interactions under an event and select a particular interaction; post comments on the discussion (i.e. interact with each other on the interaction question). The following figure shows an Interaction started under the Linden Community Development Event.
The matching part is implemented as “Keyword Match”. The user is asked to enter the Keyword s/he wishes to search. This Keyword is searched for in the User nodes. The nodes which have this keyword as their Assets are listed. Figure 29, shows a query, where a user is searching for “Users” who have computers and the list of users having computers listed as their Assets is displayed below.
8.2 Integrating With CETI Website

We plan to move the scaffolding to the CETI website; the matching function will be integrated into the Interaction page i.e. user will not have to go another page to search for the resources. We will have a match button in the Interaction page itself. Also, we plan to make the Interaction free flowing by implementing a parser that will automatically pull out the keywords making the UI friendlier. This will allow the users participating in the interaction an option to search the potential resources, still
Keyword based only, from the Interaction page rather than navigating to another page.

For example, from our example scenario, say the Interaction is “Setting up a Computer Lab”, the parser will automatically pull out the keyword-Computer and when the user clicks on the Search button, on the same Interaction page, the matched resources are shown. Stanford Language Parser will be used to extract the nouns from the sentences which in turn will be used as keywords.

8.3 Hezo Design Architecture Validation

This section shows how the Hezo system design supports the example scenario in Table 1. Here, we show how Hezo, supports the collaboration through all of its phases.

Co-discover:
Once an interaction is started i.e. a user posts the question on Hezo, the Dialog Extraction Agent extracts the keywords from the question posted. These keywords are used to suggest users/assets – based on their skills/affordances and based on the roles they have played. Hezo also suggests project semantics from previous projects. At this point, the system will also send emails to the users identified, requesting them to participate in the interactions. In case, the system fails to come up with any matches, it sends an email to the Facilitators. This human computation is necessary when a new interaction with no similarity to previous interactions is started. The facilitator then triages this interaction by using human computation to suggest users/resources.
Co-commit:
Once the suggestions are given to the users, a Coordinator is chosen from the suggested potential roles or one of the Friends, chosen dynamically from the interaction thread, is assigned to coordinate this interaction. After the coordinator is assigned, the interaction proceeds to the next stage of co-commitment. In this stage, coordinator can identify the type of interaction. If it is a clarification, he will coordinate how the interaction should proceed. The coordinator or beneficiary will be asked if the interaction is a task or not. If it is, the coordinator will 1) keep track of the requirements, 2) the status of the task and 3) Fire new interactions if required. An interaction can end here and start another interaction to proceed further or request the facilitator to administer the interaction. The participants, donors are identified by the end of this phase.

Co-create:
More roles are identified as the interaction proceeds to the next phase of co-creation. Here, the interaction is more focused i.e. the identified roles (coordinator, participant, donor, etc.) are closely involved in the interaction. In co-creation phase, the actual execution of the project takes place. If interaction is a clarification, the final sub-goals of this Interaction’s goal are discussed. If it is at task, the execution of the goal starts. The UI of the system provides mechanisms to keep track of the requirements from the earlier stage and also the status of tasks etc. An interaction can end here and start
another interaction to proceed further or request the facilitator to administer the interaction.

Co-deliver:

The Interaction is in co-delivery stage, if the goal/sub-goal is achieved. Here the interaction is successfully completed. If not coordinator can span new interactions or request facilitator to administer the interaction. If the interaction is completed the system collects feedback from the users. From a user’s level of participation and the votes of other users about their performance, the user is asked to become an ambassador for the system. After serving as an ambassador, based on his participation s/he might become the facilitator for the system.

The next two tables show how Agents work in our example scenario.
<table>
<thead>
<tr>
<th>Interaction</th>
<th>Agent</th>
<th>Services provided by Hezo</th>
<th>Building Knowledge Base(KB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advance</td>
<td>Dialog Extraction Agent</td>
<td>Extract Keywords: Linden, Community Development</td>
<td>Update the profile of users participating in the interaction based on the content they create.</td>
</tr>
<tr>
<td>Linden</td>
<td>Social Matching Agent</td>
<td>Suggest Resources based on the Goal by searching the User profiles.</td>
<td>Updates KB with the dynamically discovered roles.</td>
</tr>
<tr>
<td>Community</td>
<td>Goal Matching Agent</td>
<td>Suggest Individuals/Organizations interested in Community Development</td>
<td>Update the KB with the project semantics that are extracted from the interaction.</td>
</tr>
<tr>
<td></td>
<td>Messaging Service</td>
<td>Suggest Roles and Project Semantics by searching KB.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Suggestion: Roles from other community development Projects.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Suggestion: Need Service Learning Courses</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Roles Agent</td>
<td>Send email to suggested resources requesting them to participate in the interaction.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If Hezo shows no suggestions send email to Facilitators.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Interaction Agent</td>
<td>Assign Roles to the Users who have participated in the interaction so far.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Identified Roles: 1) Beneficiary: Linden Community 2) Friends</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintain status and hierarchy of the Interactions i.e. the sub-interactions that Users my start.</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Interaction 1

66
Need Service Learning Courses (Could be a suggestion made by Hezo or Goal discovered from the Interaction)

<table>
<thead>
<tr>
<th></th>
<th>Suggest Project Semantics. Suggestions: 1) Set Up Computer Center 2) Develop Technology Projects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal Matching Agent</td>
<td>Suggest Resources based on the Goal by searching the KB for similar projects. Suggestions: 1) CETI 2) OSU COE</td>
</tr>
<tr>
<td>Social Matching Agent</td>
<td>Send email to suggested resources requesting them to participate in the interaction. If Hezo shows no suggestions send email to Facilitators.</td>
</tr>
<tr>
<td>Messaging Service</td>
<td>Assign Roles to the Users who have participated in the interaction so far. Identified Roles: 1) Coordinator: CETI</td>
</tr>
<tr>
<td>Roles Agent</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Interaction 2
<table>
<thead>
<tr>
<th>Interaction</th>
<th>Collaboration Phases</th>
</tr>
</thead>
</table>
| Advancing Linden Community          | **Co-discovery:** None suggested, send email to facilitator  
|                                     | **Co-commitment:** Discovered the sub-goal-“ Need Service Learning Courses”                                
|                                     | New Resource Suggestions: C.E.T.I, Email sent to C.E.T.I                                                   
|                                     | Start New Interaction- “Need Service Learning Courses”                                                        
|                                     | **Co-creation:** - TBD                                                                                       
|                                     | **Co-delivery:** - TBD                                                                                       |
| Need Service Learning Courses       | **Co-discovery:**  
|                                     | Suggestions: OSU COE, Few other faculty                                                                     
|                                     | Suggestions: “Should set up Computer Lab”, “ Need Curriculum for Courses”                                    
|                                     | **Co-commitment:**  
|                                     | Discovered Sub-Goals- “Should set up Computer Lab”, “ Need Curriculum for Courses”                          
|                                     | **Co-creation:** -TBD                                                                                       
|                                     | **Co-deliver:** -TBD                                                                                         |
| Should set up Computer Lab          | **Co-discovery:** TBD                                                                                       
|                                     | Suggestions: Donors of Computers from previous projects etc.                                                
|                                     | Suggestions: “Need Computers” , “ Need Space” , “ Need Internet”                                             
|                                     | **Co-commitment:**  
|                                     | Discovered Sub-Goals: Need to find Computer Donors, Need to set up Internet,                               
|                                     | -Start interaction to find computers                                                                       
|                                     | -Start Interaction to set up Internet                                                                        
|                                     | **Co-creation:** This Interaction is Task, it fired two sub-interactions and is waiting on them to finish. Once, it is notified that both the interactions are completed; this interaction goes to next phase. 
|                                     | **Co-deliver:** Interaction is completed. Populate database, collect feedback                                |
| Need computer                       | **Co-Discovery:**  
|                                     | Suggestions: OSU Medical Center, Nationwide, Chase etc.                                                     
|                                     | **Co-Commitments:** OSU Medical Center agrees to provide free computers.                                    
|                                     | **Co-Creation:** Execute the task of bringing computers and fixing them.                                    
|                                     | Might start new Interaction to finish it                                                                     
|                                     | **Co-Delivery:** But, say computers are delivered to the lab. Hezo collects feedback and any project semantics, so that it could suggest the same in the future. Report back to the Interaction that fired this Interaction, i.e. |
| Need Internet                       | **Co-Discovery:** Suggestions: AT&T, Time Warner Cable.                                                     
|                                     | **Co-Commitment:** AT&T agrees provide internet for free.                                                    
|                                     | **Co-Creation:** Wait, till “Need Computer” finishes. Start task of setting up the free Internet connection. 
|                                     | **Co-Delivery:** All computers in the lab successfully connect to the Internet.                             
|                                     | Notify the Interaction that started.                                                                           |

Table 4: Framework Support
References


http://www.ibm.com/developerworks/rational/library/content/RationalEdge/sep04/belie


[20] Do You Want to Know? Recommending Strangers in the Enterprise

[21] Aardvark paper

[22] Ontology based context synchronization for adhoc social collaborations.
