

**GRAPH BASED REPRESENTATION OF WALDEN'S PATHS IN
DRUPAL 7**

A Thesis

by

SWATI SINGH

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee, Richard Furuta
Committee Members, Frank Shipman
Ergun Akleman
Head of Department, Dilma Da Silva

May 2015

Major Subject: Computer Science

Copyright 2015 Swati Singh

ABSTRACT

The Walden's Paths Project is used by different communities in order to create and use linear paths that link to documents available on the web. Walden's Paths has been implemented as a web application using Drupal where a web service handles the core functionality of storage and representation of a Path data structure and a user interface uses this service for authoring and browsing of the paths. I have implemented an interactive user interface for representation of information for educational purpose in the form of a graph instead of a linear representation. This representation helps the users of the path to better understand the subject by understanding the conceptual structure of subject. The authoring interface is simple and easy to use and enables the authors of Walden's Paths to represent the conceptual structure of a subject domain in the form of a graph. It allows for graphical representation of different types of relationships between various topic included in the Walden's Paths. Such graphical representation of educational resources is similar to the concept of Topic Maps. The authoring tool for the Walden's Paths has been implemented in Drupal 7. This Drupal 7 version of Walden's Paths extends its implementation from a hypertextual system to a hypermedia system by supporting different document types like images, PDF, etc.

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my advisor, Dr. Richard Furuta for his continued guidance and support throughout the course of my thesis work. He has always guided me in my work and has helped me proceed in the right direction and in shaping my thesis.

I would like to express my gratitude to Dr. Frank Shipman for his valuable suggestions and guidance in my research work. I would like to thank Dr. Ergun Akleman for taking interest in my work and for his continuous support.

I would like to thank Himanshu Barthwal and Luis Meneses for their valuable suggestions, support and cooperation.

I would also like to thank my family and friends for their unconditional love and support and for helping me achieve my Masters by constantly encouraging me.

TABLE OF CONTENTS

	Page
ABSTRACT	ii
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES.....	v
LIST OF TABLES	vi
1. INTRODUCTION.....	1
2. LITERATURE REVIEW AND RELATED WORK.....	4
3. MOTIVATION	8
4. PROBLEM STATEMENT	10
5. IMPLEMENTATION	12
5.1. User Interface.....	13
5.2. Drupal Backend	24
6. EVALUATION AND DISCUSSION.....	26
7. FUTURE WORK	30
8. CONCLUSION	32
REFERENCES.....	33

LIST OF FIGURES

FIGURE	Page
1. Interface for creating a new Walden's Path.....	15
2. Authoring Interface showing node's right click context menu.....	16
3. Authoring Interface showing addition/modification of a node content.....	18
4. Authoring Interface showing link's right click context menu	20
5. Authoring Interface showing adding of a label to an edge	20
6. Walden's Paths Viewing Tool showing the list of created path	21
7. Walden's Paths Viewing Tool illustrating a graphical representation of Walden's path	22
8. Viewer window of Viewing Tool with an online PDF document open	23
9. Window showing summary of a node element in a Walden's Path graph ...	23
10. An example showing a zoomed out image of Walden's Path	29
11. A zoomed in image of the example shown in Figure 8	29

LIST OF TABLES

TABLE	Page
1. Comparison of Strand Maps, Walden's Pathsv1 and Walden's Path v2.....	28

1. INTRODUCTION

Walden's Paths is a hypertextual system that facilitates authoring and browsing of linear paths of nodes containing information elements. Walden's Paths supports path-centric browsing that combines both graph-centric and node-centric browsing.^[1] Walden's Paths makes browsing easier for users who wish to browse resources on a specific topic in a focussed manner and hide other unrelated data available online. Walden's Paths authors can create paths that can be used by the users for browsing or for creating new customized paths from existing ones. The users browsing the Walden's Paths can choose from different paths available to them and begin browsing the chosen path. These paths can also be shared between users. A typical use case of Walden's Paths is to enhance academic learning in schools where teachers would author paths related to course work for the benefit of students. The students would then use these predefined paths to explore the course further without exposing themselves to irrelevant information that is unrelated to the course.^[2] A re-imagined Walden's Paths was suggested using a web service to isolate core functionality of path storage and representation from the user interface.^[3] A prototype of Walden's Paths was implemented in Drupal 6 and was later ported to Drupal 7. This prototype provides a simple interface to the users to create and browse Walden's Paths. This prototype focuses largely on the viewing interface for readers and on the display and use of the web documents rather than on authoring of the path and the structural complexity of the subject domain.

Hypermedia systems can be expressed as a large network of nodes where nodes represent the information entities and the edges represent the relationship between these entities. The relationship between the node entities can be diverse in terms of its attributes and nature. Some examples of these relationships are part-whole relationships, precedent topics or prerequisites, subsequents, similar or related topics, etc. These

relationship can be represented by directed or undirected edges in a graph which also affects the type of graph structure generated by the node entities. A part whole relationship can be represented using a hierarchical graph or tree structure. A graph in which the edges represent the prerequisites for understanding a topic will likely be a directed acyclic graph while a graph. A graph depicting similar or related topics where the edges represent the similarity relationship will eventually turn out to be a cyclic undirected graph. These relationships between the subject domain and its content are difficult to express by tools that support browsing in a linear fashion. Although systems like Walden's Paths support path-centric browsing, they can be enhanced by allowing better representation of paths and relationship between the information entities. Also this approach is closer to the cognitive way of learning and how the human brain processes information, which is seldom linear as opposed to current methodologies for learning. Such a graphical representation will promote manifestation of nodes in the context of their neighboring nodes as well as that of the entire path. It also imposes an author specified structure and context on the web resources and documents that is presented graphically to the users of these paths.

This thesis proposes to redefine Walden's Path to represent the contexts of the paths explicitly through a graphical manifestation of the paths instead of a pure linear approach. This thesis proposes to implement a simple graphical interface where the users would be able to author paths in a graphical format and also to identify and represent the relationship between the nodes. Creating a visual interface for authoring and browsing of information is a non-trivial task as it involves representation of highly interconnected data entities and different types of relationships among them. Although such a graphical representation offers a more cognitive learning technique to the users, it can also result in an overdose of information and can complicate learning. Thus, it is necessary to provide a simple and user friendly interface to the users where they can create paths for effective and easy learning process. This thesis also focuses on graph visualization techniques that

can be used to present the information effectively to the users in the most visually pleasing manner such that it is easier for the users to browse through the resources.

The rest of the thesis is organized as follows. Section 2 presents a literature review and overview of the existing work. Section 3 presents the motivation behind the thesis and evolution of the problem statement. Section 4 describes the problem statement in detail and section 5 elaborates on the implementation details and the work done done. Section 6 presents an evaluation of the work done and the results while section 7 elaborates on future work and ideas that can be implemented to further enhance Walden's Paths. Section 8 summarizes and concludes the work done.

2. LITERATURE REVIEW AND RELATED WORK

The Walden's Paths Project is developed and maintained by the Center for the Study of Digital Libraries at Texas A&M University. Many papers have been published on the subject that describe the motivation behind the design of Walden's paths and its evolution over the years.

[2] provides an overview of the design and initial implementation of the Walden's Paths. Walden's Paths was developed for the K-12 school environment in order to support the teacher-student relationship with the use of a Web based learning tool. The heterogeneity of the web posed a challenging problem, which was not just limited to the content and information available but also the type of readers. Path-like mechanisms were already being used to facilitate personal-space organization through bookmark lists and also to facilitate interpersonal communication through sharing of bookmark files. The paper proposes the implementation of a Web-based tool that taps the potential of paths as an effective browsing mechanism instead of path-like mechanisms. These Walden's Paths are created by the authors who are generally teachers which were later used and shared by the students. The Walden's Paths are directed paths created by the path authors using the web documents available online. These paths generally have some sort of rhetorical structure that is imposed by the content of the web documents and also by providing some sort of annotating text to the page. This ordering of web resources in the form of a path helps readers in making use of the information available online in a more focussed way without getting lost in the abundance of hypertext documents that crowd the web. Although the readers have an option of exploring the web by following the links on the pages, Walden's Paths has controls that allow the readers to return back to the path when they need to end their exploration. The paper also talks about the architecture of this

initial prototype of Walden's Paths, its implementation and experience of the users of Walden's Path with respect to authoring and browsing of paths.

[1] focusses on path-centric browsing offered by Walden's Paths and presents the design and implementation of Path Engine which is an interface tool for browsing interconnected paths in a linear fashion. Path-centric browsing combines aspects of two common browsing paradigms, viz, graph-centric browsing and node-centric browsing. Most modern day browsers support node-centric browsing where the users can interact with the node currently being displayed on the browser interface. Path-centric browsing facilitates browsing resources relevant to a particular context and also conveys information regarding the size of the hypermedia system. The paper focusses on issues that occur with generalized path structures and presents an architecture to handle these issues. It also presents issues that arise with browsing of paths that are distributed across different networks. This paper also emphasizes on the need for specification and presentation of more expressive path structures. The paper identifies that the issue with representation of complex path structures is to display it to the users effectively. The Path Engine employs a linear display strategy to keep things simple and easier for the users to use the paths. Although this prototype does not allow branching paths, the underlying architecture is capable enough to support them.

Several issues as well as new requirements were reported by the users of the initial prototype of Walden's Paths. In order to address these issues, Walden's Paths was redesigned in [3], which resulted in a new prototype for the same. The design was changed to support dynamic and branching paths. The path data structure was represented using a web service that isolated the user interface for authoring and browsing of paths from its storage. This isolation of storage from user interface via a web service allowed for multiple user interfaces like a collaborative authoring tool, a mobile path authoring and viewing tool, an offline path packaging tool and a new path

viewing tool. A prototype of Walden's Paths was implemented in Drupal 6 which presented a simple interface to the users for authoring and viewing of Walden's Paths. This prototype enables the users to create nodes that consisted of a topic and annotating text for the topic. These nodes were linked to the other nodes in the path in a linear fashion. This prototype allowed branching to some extent as the nodes could also be organized into categories which could be browsed using the upwards and downwards links. The paper presents a brief discussion of these interfaces and their prototypes. It also focusses on the results and feedback from informal studies conducted to study the experiences of path authors in creating the paths using the prototype.

Since Walden's Paths is mostly used in an educational context where a teacher is mostly an author and the students are the users, it would be interesting to think about different ways in which Walden's Paths can provide a better learning experience to the students. Concept maps have been proposed as an effective tool to facilitate learning among students^[4]. A concept map is basically a structure diagram that illustrates different concepts and the relationship between the concepts. A concept map consists of nodes that represent concepts or ideas while labelled links denote the relationship between the nodes. These are mostly used as teaching and learning tools and also for knowledge representation and organization. Poorly organized frameworks of educational materials results in limited learning and concept maps are an excellent tool to address this problem.

The use of concept maps as a powerful learning technique is supported by Ausubel's Assimilation Theory and Novak's Theory of Learning that states that individuals are expected to learn better by establishing significant connections between concepts as opposed to those who create no meaningful connections^[5]. Ausubel's assimilation theory also states that people learn new things by integrating their current knowledge with new knowledge. This theory has been the backbone of concept maps that graphically

represent this integration of concepts by illustrating the relationship between concepts as perceived by our cognitive knowledge structure. Since graphic representations are easier to understand and comprehend, concept maps have emerged as a powerful tool in collaborative learning, knowledge mapping and organization.

[6] focuses on the issues that arise with representation of structure and topic name space of learning content with respect to authoring of Topic Maps. The paper argues that a tree like representation as outlined by traditional coursework textbooks is not sufficient to represent the relationship between the content from the author's notion of classification. The hierarchy of concepts imposed by the tree like structure might not always be true.

[7] presents a Strand Maps web service interface that supports concept browsing interfaces where educators and learners can browse, locate and use the educational resources. The interface represent certain benchmarks that define what learners must know during their primary and secondary education. Strand Maps reference these benchmarks and the correlation between the ideas through a node-link representation where the links represent the relationship between the benchmarks. The relationships between the benchmarks represent background information or prior research on a particular subject that might be helpful in comprehension of another benchmark. It focuses on representing science as a network of ideas that can explored rather than a list of facts to be memorized by the learners.

3. MOTIVATION

Drupal is an open source content management framework used to organize and publish content while Walden's Paths is a hypertext content management system used to represent content in the form of a path having some rhetorical structure. Thus Drupal is an excellent tool to create a web based application for Walden's Paths. The existing Drupal 6 implementation of Walden's Paths presents a simple interface to create and view paths. A typical user might expect a much better structural representation of a content management system. The advantage of using hypertext and hypermedia documents for learning is that these are linked and facilitate navigation through documents. A graphical visualization of concepts and information leads to better understanding of those concepts. Also it facilitates reuse of paths by the path authors. For example, an instructor can extract a subpath from an existing path based on the level of depth he wants to provide for learning for the students. Thus such a representation favors better subject-domain navigation among resources and context based path reuse.

Apart from this, Drupal 6 modules are highly vulnerable to security issues as they will no longer be supported in future once Drupal 8 is released and there is no backward compatibility with previous Drupal code. There are several advantages of using Drupal 7 over the previous versions. Drupal 7 is more responsive, has HTML themes and has better support for AJAX. Drupal 7 also supports RESTful web services through Entity API and Services module to provide resource representations for all entity types. These modules provide additional capabilities like mapping additional contextual information on a request. Though they are not currently inbuilt in core, they will be included in the core from Drupal 8 and further releases. Hence upgrading Walden's Paths modules from Drupal 7 to Drupal 8 would not require many changes to the existing code. Thus, Drupal

7 provides a much better choice for creating a Walden's Path module and to enhance the user interface for authoring and viewing of paths.

4. PROBLEM STATEMENT

Representation of Walden's Paths in the form of concept maps will enable the students to browse the educational resources while establishing some sort of relationship between the concepts and the subject matter at the same time. These concept map based Walden's Paths will thus facilitate top-down browsing of the subject and help in developing better understanding of the concepts by browsing the nodes and brainstorming the associations between them rather than browsing the resources in a linear fashion. This was the motivation behind redesigning Walden's Paths to support a more cognitive approach to learning and path-centric browsing of resources available online. This thesis focusses on the design and implementation of this redesigned Walden's Paths that adopts a graphical representation instead of a linear one.

A graph based representation of Walden's Paths can better represent the conceptual structure of a subject domain. Every node in the Walden's Paths represents a piece of information or a web document. Different types of relationships can exist between these documents, for example, Part-whole relationship, similarity, etc. A visual representation of such relationships between documents helps readers in better understanding of concepts as compared to a linear representation.

A path in the Walden's Paths project has a rhetorical structure associated with it which can be expressed qualitatively. In most general cases, data representation between the nodes is in the form of a graph rather than a linear representation. My main focus is on creating a simple and efficient user interface that would allow path authors to create digital repositories of documents that can be connected in the form of a graph and would therefore support hierarchical browsing instead of linear navigation among documents. Scalability problems arise in such representations when the number of nodes in a graph

become so large that browsing the graph becomes difficult. Hence the interface must be able to support creation and browsing of large graphs in an efficient way. The layout of nodes on the webpage must scale well for small as well as large graphs as the size of the graph might vary for different authors based on their requirements.

The graphical representation is somewhat similar to Concept maps where the nodes represent the occurrences and the links between them represent the associations. The edges representing the relationship between the web resources can be directed as well as undirected and can be labelled to represent the type of relationship for example, part-whole relationship, similarity, etc. Also, with the evolution of web into a hypermedia system containing a rich diversity of media like images, and movies, it is necessary to evolve the existing version of Walden's Paths to support hypermedia documents as well. Web documents available online exists in varied formats such as pdf, doc, ppt, mp3, mp4, etc. The viewing tool of Walden's Paths must support these formats to provide a better browsing experience to the users.

5. IMPLEMENTATION

A Web Services architecture was used to design a Walden's Paths where the path data structure is represented using a web service that is isolated from the user interface. Web services are mostly used to provide some functionality on behalf of a owner which is used by the clients.^[8] This architecture supports multiple kinds of clients and different complex use cases which rely on a common functional core provided by the service layer. Such a design is highly suitable for Walden's Paths where a web based application for authoring and viewing of paths for different use cases can be created on top of service layer. Web Services is not included in the Drupal 7 core and hence middleware modules like Services and Entity API must be used to support the same.

The redesigned Waldens Path is implemented using Drupal 7 as opposed to the previous basic implementation in Drupal 6. It is only natural to switch from Drupal 6 to Drupal 7 for implementing Walden's Paths as Drupal 7 is far more powerful and flexible. Drupal 7 has a flexible architecture included in its core and also upgrading from Drupal 7 to Drupal 8 will not require drastic changes. The front end is implemented using HTML5, CSS, Javascript and JQuery. D3.js is used for graph visualization and layout of nodes in the graph. There are different Javascript libraries available for graph visualization like JSPlumb, Sigma.js, etc., but D3.js best serves the requirements like allowing zoom/pan, creating custom HTML nodes using Foreign Object, force directed layout and tree layout of nodes, etc. Google Docs Viewer Plugin is used to support viewing of different content types like youtube videos, PDF files, images, etc., in order to establish Walden's Paths as a hypermedia system rather than a purely textual one. HTML5 has transformed the web into a full-fledged application platform where users can play videos and sounds, view document, images and animations, and interact completely with the browser in a standardized manner. The power of HTML5 and several open source plugins is

harnessed to create an interactive Web application for authoring and viewing of Walden's Paths.

The implementation was divided into two parts:

- 1) Building a user friendly user interface for authoring and viewing of a Walden's Path
- 2) Integrating the user interface with the Drupal Backend.

5.1 USER INTERFACE

The Authoring Tool enables the users to create a Walden's Path by creating nodes that are connected to each other in the form of a graph. Every node in the graph represents a web resource while the links represent the type of relationship between the nodes. The links can be directed as well as undirected, depending on the type of relationship the user wants to assign between two nodes. For example, a part whole relationship can be expressed using a directed edge while similarity relationship can be expressed using an undirected edge. An author can create an individual node and then select from the various options available on the right click context menu. He can create a directed child node of a given node or an undirected related node. Also, the user can connect a node to an already existing node by creating a directed or undirected edge between them. The user also has the power to label an edge to show the type of relationship represented by the edge. The context menu also enables the user to delete or edit a particular node.

D3.js is a powerful Javascript graph visualization library that facilitates manipulation of web documents based on data. I have used D3.js to create a dynamic and interactive tools for authoring and viewing of Walden's Paths.

A node can represent different types of web resources, like an HTML page, PDF, Video, etc., depending on the resource URL. Google Docs Viewer Plugin is used to support viewing different content types.

The Drupal module for Walden's Paths consists of a user friendly interface for authoring and browsing of learning repositories. The interface consists of an Authoring tool that will enable users to create and modify a graph based Walden's Path and an Viewing tool that will enable the users to browse through the existing Walden's Path.

5.1.1 Authoring Tool

A node in a Walden's Path represents a web source available online. These resources are often linked to each other through some sort of relationship like Part-whole, similarity, prerequisite, etc. Thus, from the point of view of ontology and human understanding of these educational materials, the resources can be represented in the form of a graph where the edges between the nodes represent some sort of a relationship between them. Thus, the authoring tool must provide the authors the flexibility to represent the inherent graph structure visually.

The authoring tool must enable the authors to:

1. Create independent nodes that point to an online resource
2. Create a node in relationship to another node
3. Link two nodes together by an edge representing the relationship between them
4. Label the edges in order to visually represent the relationship
5. Click or delete the edges or modify the type of relationship represented by the edges.
6. Delete the nodes.
7. Zoom in and out of the graph

8. Pan through the graph in order to view very large graphs
9. Automatic layout and rendering of the nodes on the page to avoid collision and overlapping.
10. Identify cycles and dismiss creation of directed edges that results in cycles by throwing an alert to the author.

Figure 1 shows the interface to create a new Walden's Path and to edit and view the existing Walden's Paths.

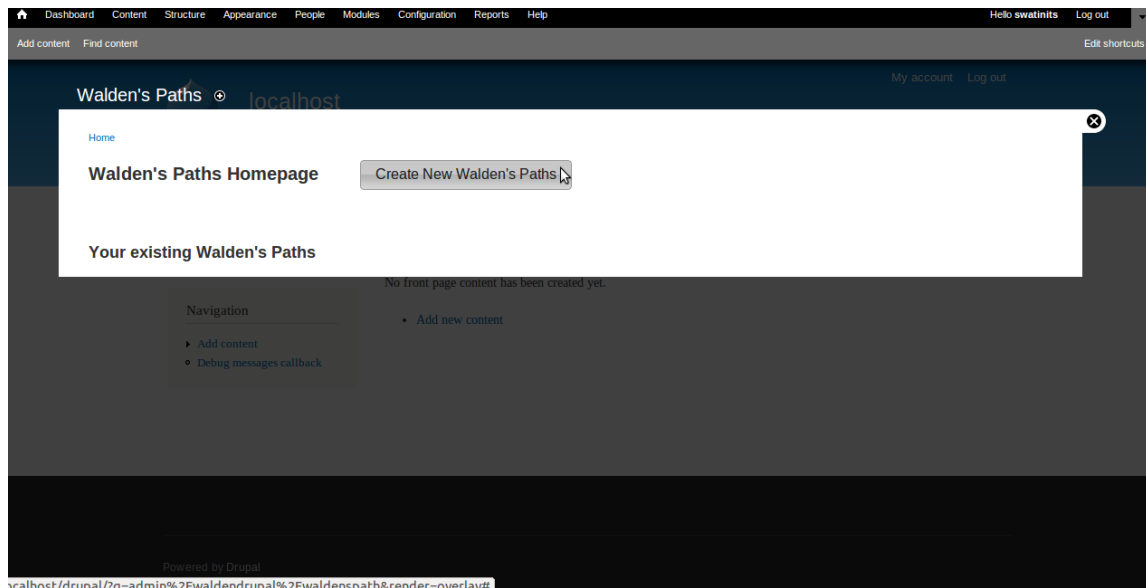


Figure 1 Interface for creating a new Walden's Path. Path authors can click on the "Create New Walden's Path" button that will guide them to the authoring interface.

There are several graph based open source libraries available to facilitate this task. I have tried and experimented using libraries like jsPlumb, Arbor.js, D3.js, etc. jsPlumb is a lightweight javascript library that enables creating connections between different elements on a webpage but doesn't have an inherent layout algorithm to place the elements in the page. It is required to manually determine the position of the nodes on

the web page. Moreover it doesn't support zooming and panning of the graph as well. Similarly arbor.js doesn't support automatic rendering of the nodes. Moreover it doesn't support clickable links in order to modify the links and the labels through the user interface. After experimenting with different javascript libraries, I found that D3.js makes it possible to support all the features expected in the authoring tool. D3.js is general purpose graph visualization library that supports SVG, Canvas and HTML elements.

Walden's Paths can be in the form of a connected as well as a disconnected graph. An author can add independent nodes to the Path that can have no edges between them. However it is impractical to create such a graph as it does not represent any inherent relationship between the nodes.

A node in the Walden's Path is a basically an HTML element that encapsulates button elements within a div element. The nodes are right click enabled which would open a context menu with different options to select for the author.

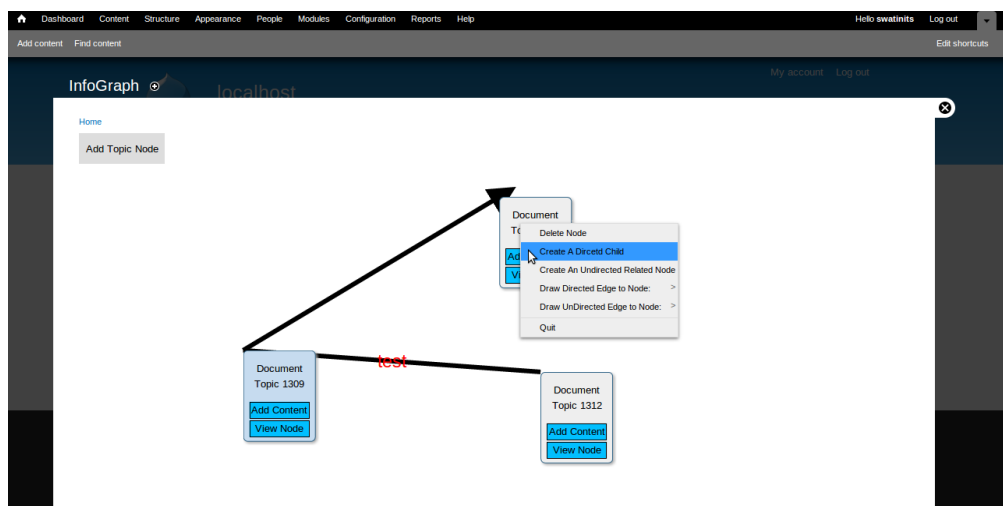


Figure 2 Authoring Interface showing node's right click context menu.

Figure 2 shows the authoring interface with a context menu that opens up on right clicking a node. Path authors can select from the list of available options on the context menu. The options available on right click context menu for a node are:

- 1) Delete the node.
- 2) Create another node with a directed edge to the node.
- 3) Create another node with an undirected edge to the node.
- 4) Link to an existing node by creating a directed edge.
- 5) Link to an existing node by creating an undirected edge.
- 6) Add and modify the content of the nodes like the URL and name of the document.

There are a number of libraries available that support creating a right click context menu. I have used jQuery contextMenu plugin^[9] to enable the context menu on right click. An interesting feature of this plugin is that it allows the context menu to be created dynamically on click. Thus menus can be created on demand and the available options can vary based on the triggering object. This was important in order to provide a user the list of available unconnected nodes for every node whenever he requires to create an edge to an existing node. Another alternative would have been to provide the user the ability to create an edge via mouse drag, i.e., by clicking on a node and dragging the pointer to an existing node. However, this would have been troublesome to the users in case of large graphs where the nodes in question would be placed at far ends of the page. Since click and drag can be inconvenient to most users for large graph, selecting a node from a list of available nodes on a context menu provides a more user friendly alternative in order to create an edge.

The nodes in the graph are rendered on the web page using the Force Layout^[10] of D3.js. Force directed graph drawing algorithms enable positioning of the nodes in the graph in a visually pleasing way such that there is no overlapping of the nodes and minimal

crossing of edges between them. The Force Layout of D3.js uses position Verlet integration^[10] to manipulate constraints like pseudo gravity force and repulsive charge force between the nodes and hence positions the nodes in the visible area. There are different parameters like size, link distance, link strength, friction, charge, etc. that can be changed depending on the size of the graph in order to represent the graph in a more aesthetically pleasing way.

The HTML element representing a node encapsulates a button that can be clicked in order to modify the contents of the node. A popup form will open upon clicking the Edit button where the author can enter the name of the document and the URL address. He can also add a text summary of the document that is being referred to by the node. Figure 3 shows addition/modification of a node content through the authoring interface.

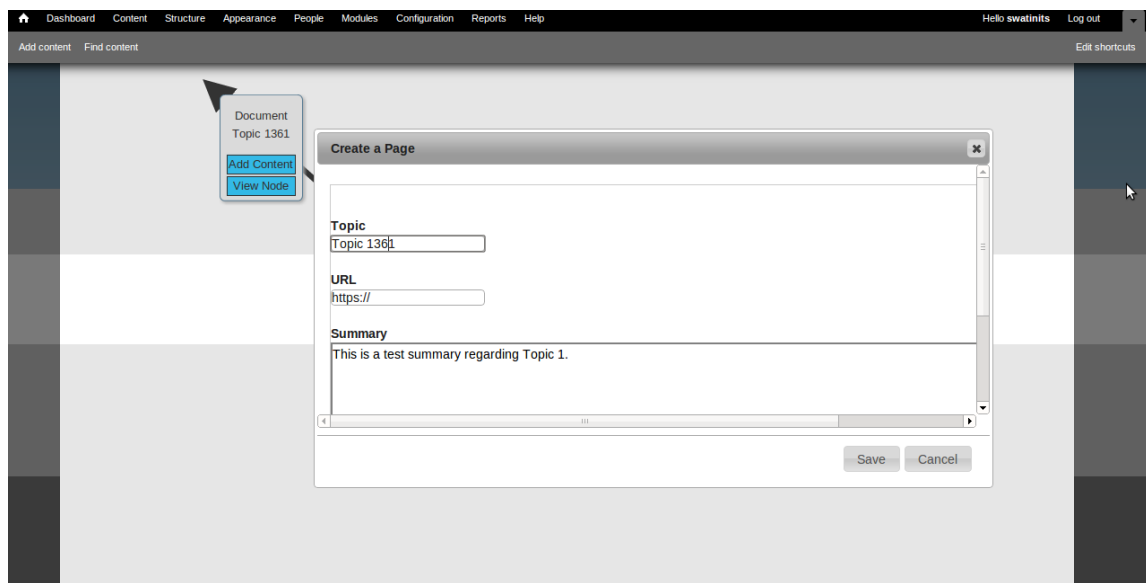


Figure 3 Authoring Interface showing addition/modification of a node content. Path authors can enter the information content of a node into the form. This includes the Document name, its web URL and annotation text regarding the document.

The edges in the graph can be directed as well as undirected. Part-whole relationships are a typical example where the author might need to create a directed edge in order to represent the topic and its subtopic. Prerequisite relationship is another example where a directed edge might help authors to represent which documents must be read prior to reading other documents for better understanding of the subject. Similarity or Related topics can be connected using an undirected edge. Cycles are not allowed in the graph in case of directed edges. Relationships that are represented by directed edges such as part-whole relationships cannot form cycles and can create ambiguity. For example, if A is a subtopic of B and B is a subtopic of C, then a directed edge from C to A would imply that A is a subtopic of C which is not practically possible. Also, it would create the ambiguity about which document represents the main topic and which are the sub-topics since there is no way of determining the head node of the cycle.

The users can also label an edge by right clicking on an edge which would also open a context menu. Figure 4 shows a context menu that opens up on right clicking a link. The context menu would provide options to the author to delete an edge or edit the label. The label would aid in clear representation of the type of relationship between the nodes in connects. Two nodes can be connected to each other through only one edge, either directed or undirected. The edges can also be labelled in order to represent the type pf relationship between the connecting nodes. Figure 5 shows the addition of edge labels by using the edit label option on the right click context menu for a link. Thus this module of Walden's Paths restricts the authors by allowing only one type of relationship between documents.

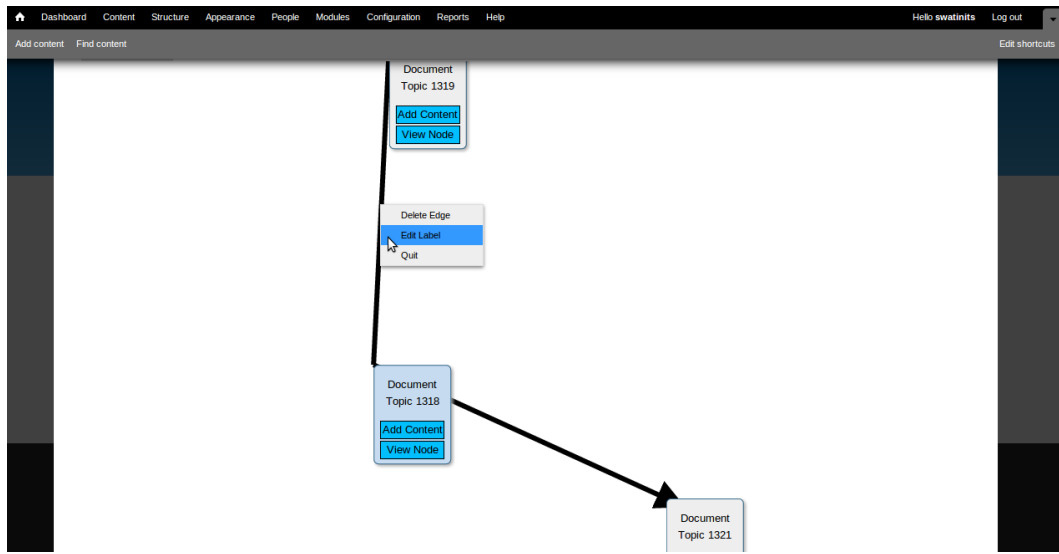


Figure 4 Authoring Interface showing link's right click context menu. Path authors can select from the list of available options on the context menu.

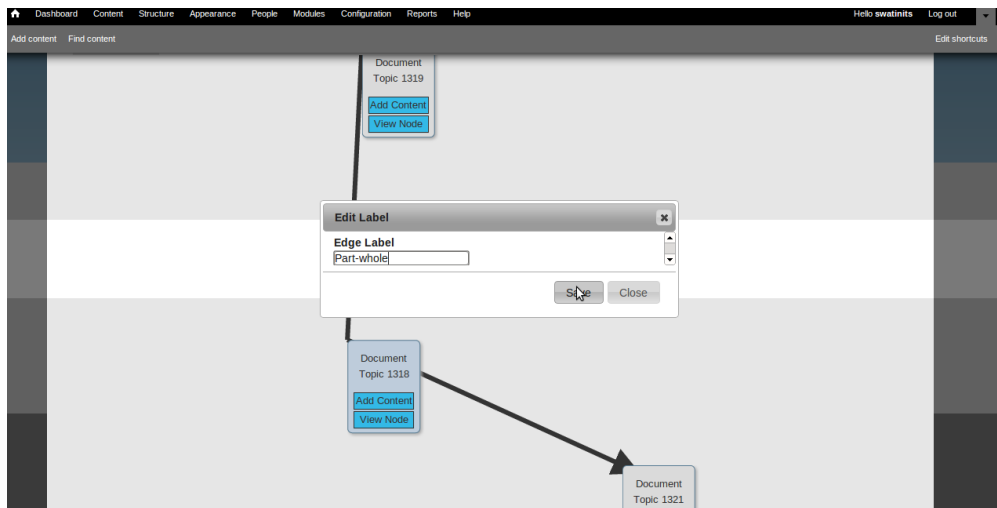


Figure 5 Authoring Interface showing adding of a label to an edge. Labels can be added to directed as well as undirected edges and are optional.

5.1.2 Viewing Tool

The Viewing tool must enable the users to:

1. Browse through the graph easily.
2. Click on the nodes to view the content that can be of different multimedia content types like video, PDF, word documents, HTML pages, etc.
3. Zooming and Panning of the graph
4. Collapsing and expanding the internal or leaf nodes of a node in order to shrink large graphs into a smaller graph and extend it back to the larger one.

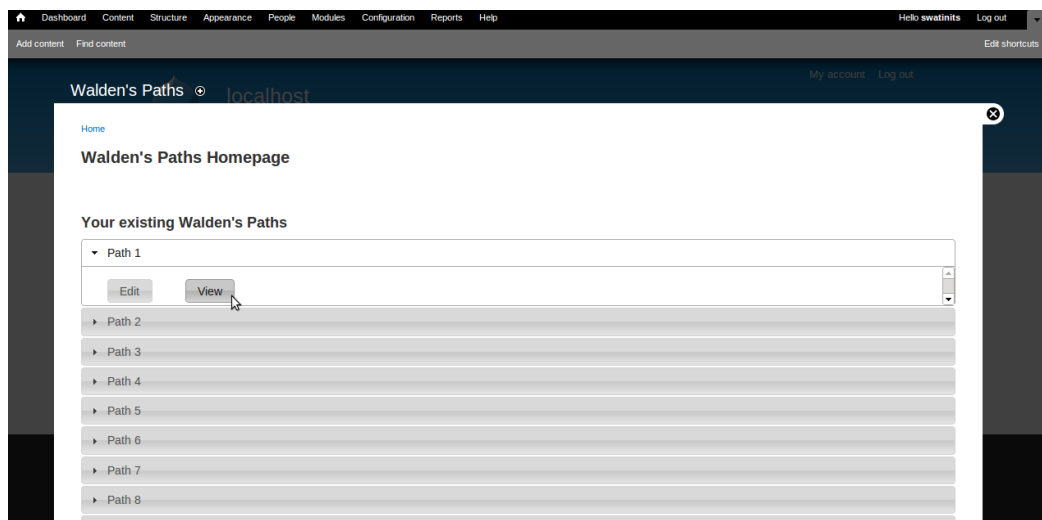


Figure 6 Walden's Paths Viewing Tool showing the list of created paths.

Figure 6 shows the list of all the created Walden's Paths. Path authors have the permission to edit as well as view the paths. The users can view a Walden's Path by clicking on the View button. The viewing tool will extract the information stored in the database and render the graph on the web page using the Force Layout of D3. Figure 7 shows the Walden's Paths Viewing Tool with a graphical representation of a previously

created Walden's Path. The Viewing Tool provides support for zooming and panning in order to support better browsing of the graph. The nodes encapsulate a view button that opens up the web document referenced by the URL of the node. I have used Google Docs Viewer Plugin in order to support different multimedia types. It currently supports video, PDF and image files. The users can collapse and expand the internal nodes. This feature is only supported for nodes whose children do not form a cycle. This feature is more useful for hierarchical tree based layouts where it helps in identifying the leaf nodes of a node. However in case of an intricately connected graph with a large number of cycles, it may not be very useful.

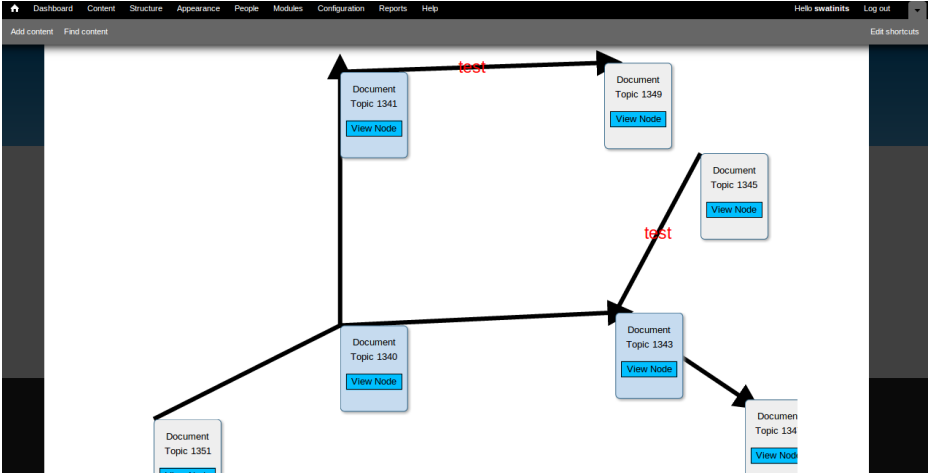


Figure 7 Walden's Paths Viewing Tool illustrating a graphical representation of Walden's Path. Both zooming and panning of the graph is allowed in order to view and browse the graph in an easier way. The user can click on the "View Node" button to view the information content represented by the graph.

The Viewing Tool enables the users to view the content of the web document referenced by URL of the nodes. Figure 8 shows a viewer window of Viewing Tool with an online

PDF document open. Figure 9 shows a window containing the text summary of a node element in a Walden's Paths graph.

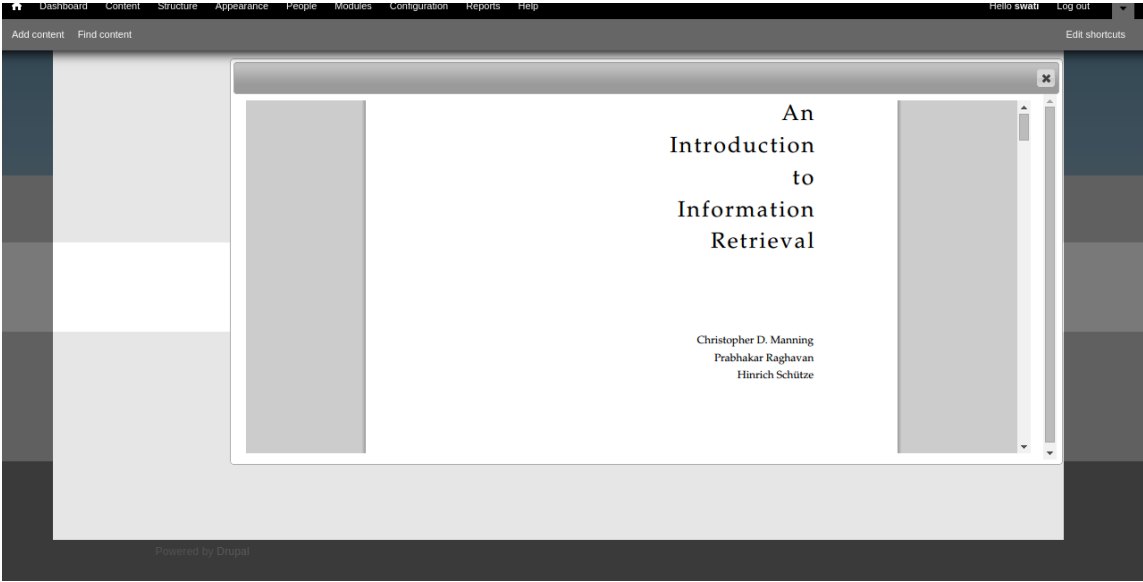


Figure 8 Viewer window of Viewing Tool with an online PDF document open.

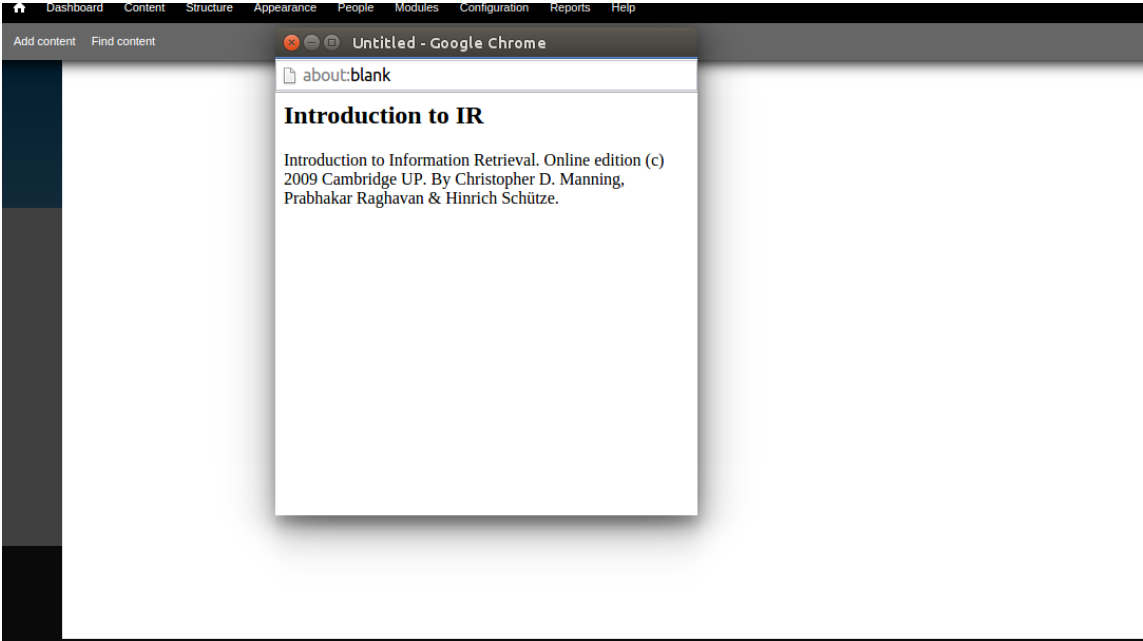


Figure 9 Window showing summary of a node element in a Walden's Paths graph.

5.2 DRUPAL BACKEND

Drupal 7 is being used to support the backend for Walden's Paths. Drupal is free and open source software that is actively maintained by a large community of developers. Drupal is widely used by developers to build content management systems like Walden's Paths.

In order to support Walden's Paths, I have used Services, Entity API and Views module along with the core Drupal 7 modules. These modules will be included in the core with the Drupal 8 release and hence upgrading Walden's Paths from Drupal 7 to Drupal 8 would be a trivial task and would require very minimal changes.

The Services module provides a way to integrate external applications with Drupal. Multiple interfaces like REST, XML RPC, JSON, etc., are supported by Services API. This module enables creation of services or collection of methods that can be used by remote applications. It also enables interaction with existing Drupal modules like nodes, views, etc. Services API is best suited for CMS Systems with Javascript frontend which is the case with Walden's Paths.

Entity API is used to create entities and bundles that are basically typed classes. An entity type can be considered as a base class whereas a bundle type is an extended class. Each of these bundles consists of fields that represent a class member. Entities are instances of the entity types. Every entity has its own Controller which is responsible for all operations on the particular entity like create, delete, edit, etc.

Walden's Paths backend consists of an infonode entity type that has three bundle types: graph node, path node and path link. The graph node bundle represents a unique

Walden's Path, i.e., there would be one graph node object for each Walden's Path. The graph node consists of a list of nodes, a list of edges and the topic name as its fields. The path node bundle represents each node in the graphical representation of a Walden's Path where a node represents a web resource. The fields for the path node bundle are graph id that represents which Walden's Path the node belongs to, the name of the document represented by the node, the web URL for the resource and a text summary for the node. The path link bundle represents each edge in the graphical representation of a Walden's Path. The fields for the path link are the source path node id, target path node id, type of edge (directed or undirected) and the label for an edge which is optional.

Thus path node and path link are basically the entity bundle representation of the nodes and edges that the users can graphically view on the web page. The file `infograph_api.js` contains the code to connect to, read from and write to the service in JSON format. I am using AJAX to make calls to Services to create and update the custom entity bundles. A REST Services endpoint is created which is used to interact with the service while making the AJAX calls.

Thus Services and Entity API modules of Drupal were used to abstract a web service for path storage and representation while a HTML5 and Javascript frontend handles the creation and nodes and edges. Various Javascript libraries were used to facilitate the task of graph layout and viewing of hypermedia resources.

6. EVALUATION AND DISCUSSION

The prototype implemented in this thesis focuses largely on the authoring interface for Walden's Paths and on the display and manipulation of path structure. It also focuses on graphical representation of relationships between the web documents that are included in a Walden's Path and the complexity of the path structure. The reader's interface is not addressed in detail. Whereas the non-Drupal implementation of Walden's Paths focuses largely on display and use of the materials rather than on the authoring of path structures.

I have performed formative evaluation of the user interface by comparing it with the existing prototypes of Walden's Paths and other prototypes that resemble Walden's Paths in nature and attributes. One such prototype is Strand Maps^[7] which is an interactive tool to explore scientific and mathematical concepts and the relationship between those concepts. I present a comparison with Drupal 6 version of Walden's Paths and Strand Maps in terms of different features available to the user.

Strand Maps aims at supporting primary and secondary educators and learners through a concept browsing interface used by digital library developers. The interface supports browsing and locating nationally-recognized learning goals or benchmarks. Concept Map visualizations represent the interconnection between these learning goals and benchmarks. The benchmarks are represented by the nodes whereas the links represent the relationship between those benchmarks. The interconnections between the benchmarks specify the background information or prior research student conceptions. SMS Javascript API and REST-based Concept Space Interchange Protocol was used to insert Strand Maps into web pages and to facilitate searching through the information space. The Strand Maps provide a useful means to organize online resources and to present a big picture of educational content and student comprehension. Strand Maps

provides a standard concept map based representation of digital library resources to facilitate learning for the learners and educators. However, its usage is restricted as it does not provide the flexibility to the educators to organize the resources as per their need. It only serves as a benchmark to browse the resources and the learning goals as defined in the Benchmarks for Science Literacy. Walden's Paths allows user to create their own graphs and paths as required by the authors and the flexibility to edit or modify them based on the requirements. This however puts the responsibility on the graph authors to enforce the relationship between the educational resources that would facilitate cognitive learning among the readers in the best possible way.

The initial prototype of Walden's Paths provides a simple interface to the users for creating and browsing Walden's Path. It does not have an advanced viewing tool to view the web resources. Neither does it support zooming and panning of the paths. Figure 10 shows a sample Walden's Path created using the Drupal 7 prototype of Walden's Paths authoring tool while Figure 11 shows a zoomed in image of the same graph. Strand Maps has an efficient browsing interface to view the topics and the relationship between them. The relationship between the nodes is limited to the prerequisites for a particular topic. Walden's Paths Drupal v2 however provides the flexibility to depict different types of relationship between the edges. It also supports creation of both directed as well as undirected edges between the nodes and labelling of edges to illustrate the relationship. Strand Maps does not have an authoring interface where users can create graphs as opposed to the prototype discussed in this thesis. Strand maps allows for panning of the graph for better browsing experience. The Drupal v2 prototype also allows for zooming in and out of the graph. Strand Maps has a mini-map at the bottom of the page that shows the portion of the graph currently being viewed in the viewport relation to the entire graph. Strand Maps also allows for viewing of relationship of a topic with respect of related topic in other maps. This feature is not yet supported by Walden's Path Drupal v2 prototype. Table 1 lists all the main highlights of the results of

the comparison study of different features available with Strand Maps, Walden's Paths Drupal v1 and Walden's Paths Drupal v2.

Table 1 Comparison of Strand Maps, Walden's Paths v1 and Walden's Path v2

	Strand Maps	Walden's Paths Drupal v1	Waldens Paths Drupal v2
Interactive interface for authoring	No	Yes	Yes
Representation of relationship between documents	Yes (single)	No	Yes
Zooming	No	No	Yes
Panning	Yes	No	Yes
Edge labelling	No	No	Yes
Mini-map	Yes	No	No
Representation of relationship with other maps	Yes	No	No

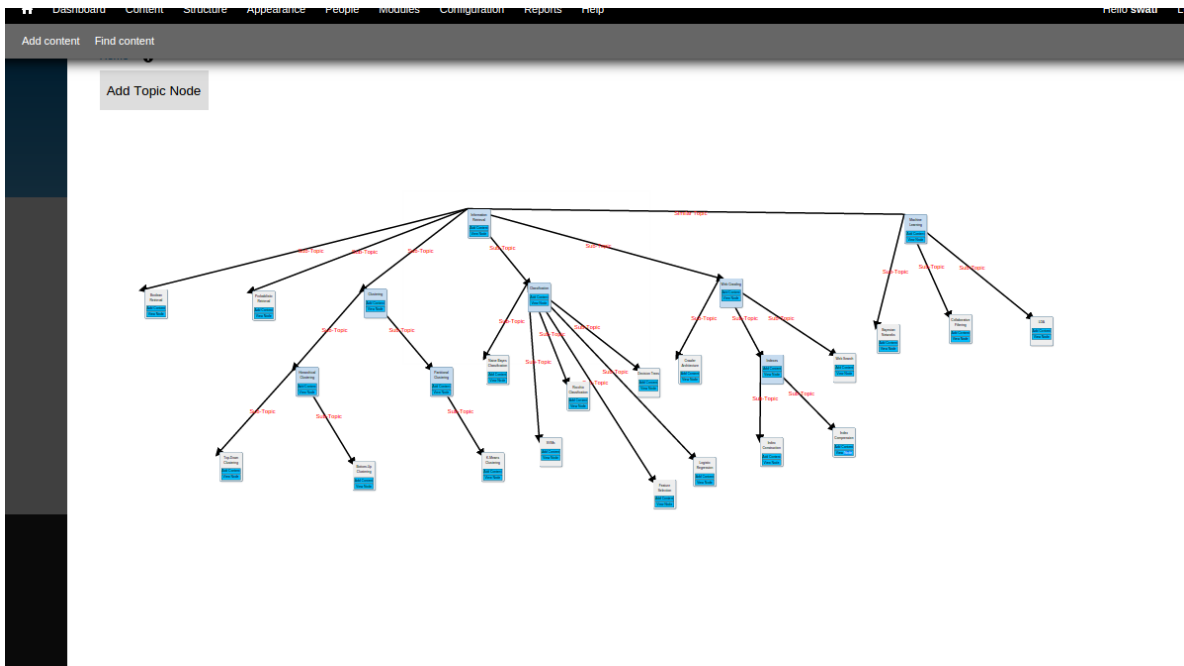


Figure 10: An example showing a zoomed out image of Waldens Path. The nodes represent various topics in Information Retrieval and related topics in Machine learning and the relationship between the topics(Sub-Topic, Similar Topics) is represented by the links.

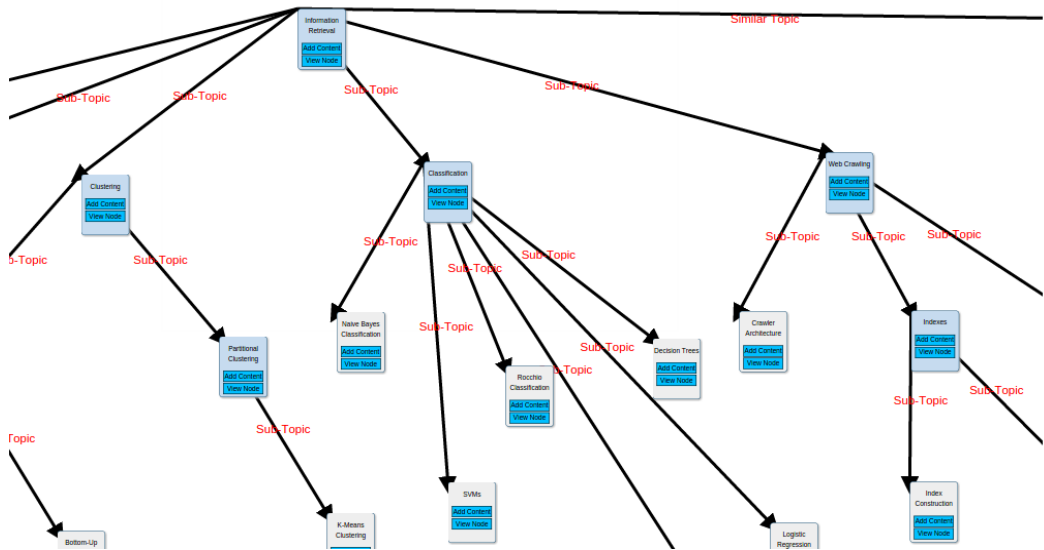


Figure 11: A zoomed in image of the example shown in Figure 8. Users can browse or focus on certain topics and later zoom out or use graph panning to drag the graph in order to browse topics outside the viewing window.

7. FUTURE WORK

The Drupal module of Walden's Paths proposed in this paper is an initial prototype for representing the relationship between the web resources in order to facilitate path-centric browsing in a more cognitive way. Several enhancements can be made to this prototype in order to provide a better experience to the users.

The prototype discussed in this paper provides a general purpose interface to the authors to create a graphical representation of Walden's Paths. The prototype can be tailored to represent a particular type of relationship between the nodes. For example, representing part whole relationship between the documents will render the graph in the form of a hierarchical tree structure or a directed acyclic graph. Thus a tree based representation would be more simpler and accurate with respect to both authoring and browsing the Walden's Paths. For directed acyclic graph, other layout algorithms like Layered graph drawing would be more efficient rather than Force Directed graph drawing. Similarly, some users might want to create Walden's Paths that supports linear browsing of web documents. Thus the current prototype can be extended to support Walden's Paths in different representation formats, like linear, tree representation etc.

The graph based representation becomes somewhat cluttered for a larger network of nodes. Thus a better layout algorithm can be implemented for placing the nodes in a more visually pleasing way and to minimize the crossing of edges. However this is a complicated problem and active research is being done in the field of representation of large graphs to solve this problem. It would be interesting to experiment with other graph based javascript libraries and other layout algorithms and to analyse if they can represent a larger Walden's Paths in a more efficient way. D3 uses Force directed graph drawing algorithm for automatic rendering of node positions in the graph. A minimap can also be

shown on the webpage where the viewport would show the position of the graph currently being viewed relative to the entire graph.

The viewing tool currently supports viewing web pages, video resources, image and PDF. This can be extended to support other document types like word documents, presentation files, etc. Web documents are available online in so many varied formats and an efficient hypermedia system must be able to support most file types in order to provide a more general purpose viewing tool to the users.

The current prototype supports automatic rendering of nodes as determined by D3's Force Directed Layout. The author has no control over placement of the nodes which changes as and when then the nodes are added or removed to the graph. Since the position of nodes in the graph is not fixed, the users might find it a difficult to browse such a graphical hypermedia system if the node positioning is important for them. Thus, a future enhancement would be to provide the users the flexibility to fix certain nodes when required and to load the Walden's Paths for viewing with the same node positioning as specified by the author while creating the paths.

Several other interfaces can be designed on top of the Drupal backend like a recommendation interface that provides suggestions for web resources for addition of new nodes or replacement of missing documents based on the existing rhetoric structure of a path, an interface to switch a path representation from a graphical form to a linear representation and vice versa, etc. The prototype discussed in this thesis is only a first step towards representation of a more complex Walden's Paths in a simpler way. It is a friendly and interactive interface for the users that would enable them to author and view paths in order to facilitate better cognitive learning process and more efficient path-centric browsing of resources. The design and architecture will only evolve based on the feedback received from users and as the requirement evolve over time.

8. CONCLUSION

This thesis is a first step towards representation of complex relationships between various web resources available for academic learning through Walden's Paths. It focusses on providing a better path-centric browsing experience to the users by visually representing the graph structure formed by the resources and the relationship between them. It helps in manifestation of the entire context of the path as a whole as well as that of each node with respect to its related nodes in the form of a graph. The prototype developed tries to provide a simple and easy to use interface to the users for the purpose of authoring and browsing of paths. Drupal's RESTful Services and Entity API helps in abstracting the storage of the path in the form of a web service. JavaScript and jQuery libraries like D3.js, jQuery contextMenu plugin, Google Docs Viewer Plugin, etc. are powerful tools that have been used to enhance the front end interface available to the users of Walden's Paths. Although authoring of paths is slightly complex with graph based representation of documents rather than a linear one, it allows for a greater flexibility in representing the relationships between web resources and also aids in better understanding of concepts from the point of view of cognitive learning.

REFERENCES

- [1] Pratik Dave, Unmil P. Karadkar, Richard Furuta, Luis Francisco-Revilla, Frank Shipman, Suvendu Dash, Zubin Dalal. *Browsing Intricately Interconnected Paths*. In: HYPERTEXT '03 Proceedings of the fourteenth ACM conference on Hypertext and hypermedia, New York, NY, USA, 2003.
- [2] Richard Furuta, Frank M, Shipman, Catherine C. Marshall, Donald Brenner and Hao-wei Hsieh. *Hypertext Paths and the World-Wide Web: Experiences with Walden's Paths*. In: HYPERTEXT '97 Proceedings of the eighth ACM conference on Hypertext, New York, NY, USA, 1997.
- [3] Paul Logasa Bogen II, Daniel Pogue, Faryaneh Poursardar, Yuangling Li, Richard Furuta, and Frank Shipman. *WPv4: A Re-imagined Walden's Paths to Support Diverse User Communities*. In: JCDL '11 Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries, New York, NY, USA, June 2011 .
- [4] Dawn M. Zimmaro, Stephen M. Zappe, Jay T. Parkes, Hoi K. Suen. *Validation of Concept Maps as a Representation of Structural Knowledge*. In: Annual Meeting of the American Educational Research Association, Montreal, Canada, April, 1999
- [5] Ausubel, D. P. Educational psychology: A cognitive view. New York: Holt, Rinehart, and Winston, 1968.
- [6] Darina Dicheva, Christo Dichev. *Authoring Educational Topic Maps: Can We Make It Easier?* In: Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies, Washington, DC, USA, July 2005.

[7] Tamara Sumner, Faisal Ahmad, Sonal Bhushan, Qianyi Gu, Francis Molina, Stedman Willard, Michael Wright, Lynne Davis, and Greg Janee. *A Web Service Interface for Creating Concept Browsing Interfaces*. D-Lib Magazine, November 2004.

[8] David Booth, Hugo Haas, Francis McCabe, Eric Newcomer, Michael Champion, Chris Ferris, David Orchard. [Web Services Architecture](#). February 2004.

[9] Rodney Rehm, Christian Baartse, Addy Osmani. [jQuery contextMenu](#), 2013.

[10] Jason Davies. [Github Wiki Reference for Force Directed Layout in D3.js](#), February 2015.

Supplemental Sources Consulted

[11] Rodney Rehm, Christian Baartse, Addy Osmani. [Documentation on jQuery contextMenu](#). June 2007.

[12] Darina Dicheva, Christo Dichev, Dandan Wang. *Visualizing Topic Maps for e-Learning*. IN: Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05), Washington, DC, USA, July 2005.