Nexus

A Framework to Derive Human Intent from Architectural Space
to Enable Context-Aware Information Exchanges in a Wireless Intranet

by
Napier Sandford Fuller
A.B., History of Art and Archeology
Washington University in St. Louis, 1996

Submitted to the Department of Urban Studies and Planning
in the School of Architecture and Planning
at the Massachusetts Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science
February 2, 2004

© 2004 Napier Sandford Fuller, all rights reserved.
The author hereby grants MIT permission to reproduce and distribute
this thesis publicly by paper and electronic means.

Author

Department of Urban Studies and Planning
Student

Certified by

Professor Joseph Ferreira
Thesis Supervisor

Accepted by

Professor Dennis Frenchman
Chairman, MCP Committee

MapNexus
A Framework to Derive Human Intent from Architectural Space to Enable Context-Aware Information Exchanges in a Wireless Intranet

by
Napier Sandford Fuller
in Partial Fulfillment of the Requirements for the Degree of Master of Science
February 2, 2004

ABSTRACT:

Architectural interior spaces provide a rich syntax for context-aware modeling. In a wireless wide area network (WAN), an urban environment can be geographically tessellated into a series of polygons reflecting the radio transmission range of each wireless access point (AP). By using these APs as spatial aggregation units in a relational database, logical assumptions about short-term behavioral patterns can be modelled; this paper describes a method to capture, encode, and interpret context-aware cues. These cues are then utilized by an intranet web server to produce context-aware output, information that fits the client's short-term activity landscape and intentions. The method does not require any special client-side software and preserves the user's anonymity as it derives the client's physical location implicitly. Such a framework permits a new type of context-aware web interaction based upon the logical "common sense" patterns that are specific to architectural interior spaces at a given time.
ACKNOWLEDGEMENTS

ECCO LETTORE.
C’È SOLO PERSONAGGI E POSTI IN QUESTA VITA.
HO MULTI BONI GIORNI PERCHÉ:

Gli Eroditi
JOE FERREIRA
TED SELKER
JOHN CLEAVER

Le Muse
MALGORZATA MLECZKO
ANDREA KULAS
ΣAPA ΡΩΣΔΕΡ

Gli Amici
JASPER GOLDMAN
JAMES WHITE
FLORIAN SEUBERT

La Familia
MARTHA NAPIER
ROSALIND FULLER
BILL FULLER
WILLIAM FULLER

Il Ambiente
CAMBRIDGE, MASSACHUSETTS
CAMBRIDGE, ENGLAND
BRUSSELS, BELGIUM
WILMINGTON, NORTH CAROLINA

Anno Domino
MMIII

MapNexus
## CONTENTS

List of Figures ................................................................. 5

Table of Abbreviations ......................................................... 6

1 Mapping Human Behavioral Patterns with Architectural Space
   1.1 The Goal to Improve Upon the “Page” .................................. 7
   1.2 The Problem of Context in Architectural Space ...................... 8
   1.3 Brief Vision of a Future Prototype .................................. 9
   1.4 Theoretical Origins, Three Strands .................................. 9
   1.5 Technical Description of MIT’s Wireless Environment ............ 11
   1.6 Overview of Thesis and Diagrams ................................... 12

2 Creating Context-Aware Web Interaction
   2.1 Example 1, Public Transportation .................................. 13
   2.2 Example 2, the Power of Suggestion ................................ 15
   2.3 Example 3, Cognition of the Spatial Environment .................. 16

3 Understanding the Context-Aware Design Approach
   3.1 Introducing Context-Aware Computing ................................ 19
   3.2 Network-Determined Context-Awareness ............................. 21
   3.3 Location-Based Context-Awareness .................................. 22
   3.4 Intent-Based Context-Awareness ..................................... 24

4 Structuring a Grammar for Human Behavior and Architectural Space
   4.1 Micro-Scaled Zoning Ordinance ..................................... 27
   4.2 Wittgenstein’s Language Games and Misperceptions ............... 28
   4.3 Minsky’s “Society of the Mind” Approach ............................ 29
   4.4 Natural Language Processing: 12-Dimensions of Verbal Context ... 29
   4.5 Determining the Five-Dimensions of Architectural Spaces ........ 24

5 Capturing Context in Real Time Information Exchange
   5.1 Presenting the Role of the Middle-Ware ............................. 32
   5.2 AP Polygon as a New Unit Spatial Aggregation ..................... 35
   5.3 Harnessing the Latent Data in the “Get” Command .................. 37
   5.4 Representing Context in a Relational Database .................... 39
   5.5 Scale of the SQL Operations ......................................... 43

6 Designing a Context-Aware Web Server
   6.1 Processing Context to Event Matches Profiles ..................... 45
   6.2 Client to Event Preference Vectors ................................ 50
   6.3 Illustrations of Context .............................................. 52
   6.4 Site Maintenance and Correction ................................... 53

7 Evaluating the MapNexus Approach
   7.1 Summary ........................................................................ 55
   7.2 The MapNexus Frameworks’s Advantages ............................. 56
   7.3 The MapNexus Frameworks’s Disadvantages ......................... 57
   7.4 Conclusions ................................................................... 58

References ............................................................................ 59

MapNexus
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MapNexus Front-End, Middle-Ware, Back-End</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Wi-Fi Coverage Areas and Crossover</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>Harvard to MIT Routes</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>Project Voyager Diagram</td>
<td>24</td>
</tr>
<tr>
<td>5</td>
<td>Zoning and Table of Use Patterns</td>
<td>27</td>
</tr>
<tr>
<td>6</td>
<td>Multi Dimensional Model for Context Markup</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>Middle-Ware</td>
<td>34</td>
</tr>
<tr>
<td>8</td>
<td>Site-Specific APs</td>
<td>36</td>
</tr>
<tr>
<td>9</td>
<td>Visual Traceroute and wi-fi Packet Analysis</td>
<td>38</td>
</tr>
<tr>
<td>10</td>
<td>Context-Aware Web Page Generation</td>
<td>40</td>
</tr>
<tr>
<td>11</td>
<td>MapNexus database and SQL Sketch</td>
<td>42</td>
</tr>
<tr>
<td>12</td>
<td>User GUI from MapNexus Scenario #1</td>
<td>45</td>
</tr>
<tr>
<td>13</td>
<td>User GUI from MapNexus Scenario #2</td>
<td>48</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>2G</td>
<td>Second Generation wireless mobile phone protocols</td>
<td></td>
</tr>
<tr>
<td>3G</td>
<td>Third Generation wireless mobile phone protocols</td>
<td></td>
</tr>
<tr>
<td>AI</td>
<td>Artificial Intelligence</td>
<td></td>
</tr>
<tr>
<td>AP</td>
<td>wireless Access Point: transmits and receives signals in a wi-fi WAN</td>
<td></td>
</tr>
<tr>
<td>CYC</td>
<td>pronounced &quot;pysche,&quot; a natural language processing method</td>
<td></td>
</tr>
<tr>
<td>DARPA</td>
<td>Defense Advanced Research Projects Agency (USA)</td>
<td></td>
</tr>
<tr>
<td>DHCP</td>
<td>Dynamic Host Configuration Protocol: assigns temporary IP addresses for</td>
<td></td>
</tr>
<tr>
<td>DSS</td>
<td>Decision Support Systems</td>
<td></td>
</tr>
<tr>
<td>DSSS</td>
<td>Direct-Sequence Spread Spectrum</td>
<td></td>
</tr>
<tr>
<td>ESRI</td>
<td>Environmental Systems Research Institute</td>
<td></td>
</tr>
<tr>
<td>EU</td>
<td>European Union</td>
<td></td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communications Commission (USA)</td>
<td></td>
</tr>
<tr>
<td>GML</td>
<td>Geography Markup Language</td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
<td></td>
</tr>
<tr>
<td>GSD</td>
<td>Harvard Graduate School of Design</td>
<td></td>
</tr>
<tr>
<td>GSM</td>
<td>Global System for Mobile communications: an EU radiocommunications protocol</td>
<td></td>
</tr>
<tr>
<td>HTML</td>
<td>HyperText Markup Language</td>
<td></td>
</tr>
<tr>
<td>HTTP</td>
<td>HyperText Transfer Protocol</td>
<td></td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
<td></td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
<td></td>
</tr>
<tr>
<td>ISP</td>
<td>Internet Service Provider: access to the internet via cable or telephone lines</td>
<td></td>
</tr>
<tr>
<td>JITIR</td>
<td>Just In Time Information Retrieval</td>
<td></td>
</tr>
<tr>
<td>LBMS</td>
<td>Location Beaconing Methods and System (from a Microsoft patent)</td>
<td></td>
</tr>
<tr>
<td>LBS</td>
<td>Location Based Services</td>
<td></td>
</tr>
<tr>
<td>MBTA</td>
<td>Metro Boston Transit Authority</td>
<td></td>
</tr>
<tr>
<td>NLP</td>
<td>Natural Language Processing</td>
<td></td>
</tr>
<tr>
<td>PDA</td>
<td>Personal Digital Assistant</td>
<td></td>
</tr>
<tr>
<td>RDBMS</td>
<td>Relational Database Management System</td>
<td></td>
</tr>
<tr>
<td>UNIX</td>
<td>Operating system that originated at Bell Labs in 1969</td>
<td></td>
</tr>
<tr>
<td>URL</td>
<td>Uniform Resource Locator</td>
<td></td>
</tr>
<tr>
<td>UTC</td>
<td>Universal Time Coordinated: the same as Greenwich Mean Time</td>
<td></td>
</tr>
<tr>
<td>WAN</td>
<td>Wireless Area Network</td>
<td></td>
</tr>
<tr>
<td>wi-fi</td>
<td>wireless fidelity: this is another name for IEEE 802.11b</td>
<td></td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
<td></td>
</tr>
</tbody>
</table>
1.0 Mapping Human Behavioral Patterns with Architectural Space

"The loss of the subject in the writing has never been more complete (the subject becoming totally irreparable) than in these utterances where the disconnection of the utterance occurs ad infinitum, without a brake, on the model of the game of topping hands or of 'rock, scissors, paper' – texts whose 'ridiculousness' or 'stupidity' is based upon no certain utterance and over which, consequently, the reader can never gain any advantage... What reader can hope to dominate such an utterance – adopt it as a laughable or critical object, dictate to it, in a word? – in the name of what other language?"

Roland Barthes, 1971, Sade, Fourier, Loyola.

1.1 The Goal to Improve upon the Concept of “Page”

Since Mesopotamian times, the structure of the written “page” has not changed much over time (Crawford, 2001). A paper page contains text (written, typed, or computer-generated) which has been created by an author in the past and which is read by an individual in the present. Hence, the relationship between the author and the reader is one of estrangement as they are separated in both time and place: a text is always removed from its original context. Nowhere is this separation more acute than within the Web. The author’s audience is largely unknowable – a reader of a web page could theoretically speak any language, live in any country, and be of any age. Moreover, the reader is increasingly not even a human being but may be a search engine (a text-parsing algorithm) or a computer aid for the visually impaired.

Since there is no defined simulacrum of “reader,” there is no conversational exchange as the very idea of “context” has been completely eroded. As the structure of the Web permits an increasingly diverse audience, this “unknowability” has become problematic in information exchange. MapNexus seeks to mimic aspects of common sense reasoning that underpins vernacular spoken exchanges; since common sense is by its nature implicitly understood by both parties in a conversation, so too will the MapNexus server be able to automatically offer content germane to the physical setting and likely intent of the client.

This paper lays out a new and unobvious framework and related methods to model MapNexus
a client's context and to produce a context-aware web page that offers information that is germane to one's immediate physical and behavioral environment. By carefully modeling the context of the information seeker in a given architectural space, a web site providing local information can offer context-aware output by making the content of a web page a function of the client's context. This thesis project is concerned with web-based information exchanges that occur in a campus-like computing environment or a wide area network (WAN) using wireless access. More specifically, this project uses the MIT campus and its students as the intended environment for implementation of a prototype; the term MapNexus is used to indicate a future prototype set in MIT's computing environment. However, this thesis project is limited to the presentation of a conceptual framework – the project has not yet been put into practice, but efforts are ongoing to build a prototype at present.

1.2 The Problem of Context in Architectural Space

"Contextere" is a Latin verb meaning "to weave together." When people speak to one another in person, the communication exchange is rich in implicit messages: body language and eye contact. Moreover, the physical environment – a crowded subway car or a private office – has innate characteristics that implicitly guide the thoughts and conversations that occur within that particular architectural space. Likewise information seeking starts with an internal conversation – talking with oneself. By looking carefully at the architectural space, one can gain cues about which conversations would be appropriate in a given space and which conversations would be unusual. MIT as a physical environment is rich in its different architectural syntax – most buildings have a specialized function. There are clearly land-use patterns within the 152-acre campus that can be modelled on a building-by-building basis.

Context-aware computer applications emphasize the relationship between the individual and the physical and behavioral setting (Selker, 2000). Context-aware applications try to ascertain an individual's intent as it relates to architectural space and the behavioral
patterns expected within in a given space at a given time. Rarely is the relationship between an individual and one’s circumscribed geographic range of activities easy to describe, yet it can be said with some certainty that one does tend to pray in the chapel, sleep in the dorm, and study in the library. There is a definite relationship between architectural space and the range of human behavior expected within the walls of a particular building. Thus the identity of the actual individuals in a space is not a part of context in the MapNexus framework, but the type of character that would be associated with a given space. The “potential energy” of an architectural space consists of things, people, and ensuing possibilities.

1.3 A Brief Vision of a Future Prototype

As indicated in Figure 1, MapNexus is a framework to capture, encode, and interpret context-aware cues about people’s anticipated behavior as a function of site-specific architectural space. When human-computer information exchanges reflect the behavioral context of a particular setting, a client saves time and is better able to make decisions in the field. Cues from a particular architectural space are utilized by an intranet web server to establish a behavioral context and a physical setting to gage one’s short-term intent.

A framework for the operation of a context-aware web server focuses upon defining the client’s short-term possibilities – the web server’s content thus becomes context-aware. Finally, a six dimension attribute model is presented to track the context of each wireless access point’s surroundings in a wide area network; this model is also used to model events in order to match them with a particular context. MapNexus is a framework that consists of three
separate parts: the client’s browser application (front-end), the context-aware profiles of data derived from the client’s location (middle-ware), and the subsequent content offered by the intranet server (the back-end).

1.4 Theoretical Origins, Three Strands

The development of MapNexus has involved the integration of three separate concepts: the spatial aggregation unit (from geography), the context-aware engine (from artificial intelligence), and natural language processing (from linguistics). The “spatial aggregation unit” or spatial aggregation unit, a concept utilized in geographical information systems (GIS) modeling, is a polygon of physical space – such as a room, a ZIP code, a land parcel, a river, or a nation – that is represented in a database as an entity. Each entity has a set of attributes. MapNexus utilizes spatial aggregation units as polygons of demarcation to identify the physical environment of the information seeker and to then draw upon context to produce web content germane to the particular time and place of the client. Specifically, MapNexus envisages the use of computers within a wireless network that act to create the spatial aggregation units around the MIT campus.

To identify the user’s particular spatial aggregation unit (i.e., location), the MapNexus server will trace the routing of the client’s first request within a browser; by performing a “traceroute” operation, the client’s location within a building can be derived from the wireless protocols of the MIT campus. This added information will allow MapNexus to create a context profile for individuals in each building over time. This approach will enable the MapNexus web server to identify far more accurately the user’s likely needs without the cumbersome process of the user having to explicitly communicate one’s location to define a context. MapNexus is thus concerned with the field of Human Computer Interaction by building a more intelligent web server.

MapNexus seeks to borrow some of the concepts to define conversational context from Natural Language Processing (NLP) in order to create a more conversational style of information exchange that is cognizant of the user’s context – one’s disposition and realm
of possibilities. Having too much information (a problem ubiquitous to the web) is like having a box of vocabulary cards in which we do not know all the meanings – it would be hard to make new sentences that are grammatically correct if we place words we “sort of know” into one sentence. This box of vocabulary cards is a metaphor for the vast quantity of information on the web, for the inquiry is often in the active first person: “I want to ____,” or “When does ____ occur?” In time/place inquiries, we often have difficulties drawing lucid relationships between what we know and new information we have found. Seeking and making sense of the vast amount of information on the web requires a cognitive ability to place the pieces of the puzzle together in order to complete the task. The time barrier to the retrieval of useful information is one of the most important determinates in a web site’s long-term success fractions of a second count. The utility of a web site often has an inverse relationship to the amount of explicit input (i.e., clicks and text) required to gather data; less is more (Huberman, 2001).

1.5 Technical Description of MIT’s Wireless Framework

MIT has approximately 4,000 people using wireless laptops on a given day; the Institute has one of the largest wireless wide area networks (WAN) of any university, with over 450 access points (APs) in its 152 acre campus. Nearly all of MIT’s buildings are easy to model in terms of context as there are no mixed-use facilities. The wireless WAN runs on the IEEE 802.11 protocol, and the wireless WAN is able to achieve a high rate of data transfer by the use of direct signaling spread spectrum (DSSS) – a physical layer technology involving bit transfer streams to be spread over a number of frequencies (~ 2.4 GHz) simultaneously.

The range of coverage of a given AP varies – by default the client communicates to the AP with the strongest signal. Most APs are within 50M of the client on the campus. The AP placement is designed to maximize the coverage in a particular architectural space. Thus the particular AP that is receiving the client’s communications is a cue to location. However,
due to obstacles to radio frequencies (i.e., walls and building materials), it is occasionally possible that a client near a window receives a stronger signal from a neighboring building (Figure 2). In general, the relation to client with the AP is one of proximity since the radio frequencies are being processed by the AP into fixed line data transmissions. Each AP has a unique address on the fixed line Internet: an Internet protocol (IP) address.

1.6 Overview of Thesis

This Chapter sets out to introduce the MapNexus project along with its potential use and inspiration. Chapter Two will seek to explain the potential use for MapNexus in a few illustrative examples. Chapter Three will explore the current status of research in context-aware computing by describing precedents relevant to MapNexus in academia and in terms of patents. Chapter Four will provide a background into the three key ideas that led to the MapNexus framework: (1) real estate zoning as a function of the spatial aggregation unit, (2) Minsky's and Lenant's formal approaches to context markup in natural language processing algorithms to create a common sense inference engine, and (3) studies of human-computer interaction regarding web-based information seeking. Chapter Five will detail a method to code context aware information as a function of site specific architectural space. Chapter Six will examine the role of the back-end web server in aggregating context to produce a context-aware web page containing content germane to the client’s location and likely intent. Chapter Seven concludes with a discussion of the relative positive and negative aspects of MapNexus and its applicability beyond MIT.
2.0 Creating Context-Aware Web Interaction

"My goal is to build a cognitive architecture based on Marvin Minsky's 'Society of Mind' theory. I think of a cognitive architecture as a system that consolidates the cognitive functionality required for ordinary commonsense problem solving. Here are some of the functions that must be served: Recognizing, Abstracting, Comparing, Expecting, Learning, Predicting, Explaining, Planning."


2.1 Example 1, Transportation Schedules

MapNexus represents a framework of information about one's immediate situational landscape; the "generic MIT student" is the intended client for the application. Hence, the design makes some assumptions about the user's behavioral patterns and means of transport. In this chapter, a few sketches of human-machine interaction in a mobile setting will be presented using the envisioned context-aware web server for MIT (called MapNexus for clarity, as it is site specific). Note that these sketches are fictitious, and the true tests of such a framework lie in their utility to the student over a sustained period of time. The goal is to encourage the client's use and re-use to maximize productivity in a given "event landscape."

Ashley takes a course at Harvard University, and she is also taking a MIT GIS class that is difficult to attend given her busy schedule. The problem for Ashley is which route is the fastest method to get back to Harvard Square in a given context (place, time) after emerging from a class in this unfamiliar place? The location of the class is not easy to define in advance as the course involves set lectures, labs, and optional help sessions in a number of locations on the main Campus of MIT (Figure 6). Ashley travels to and from Harvard Square which is about 1.5 miles north of MIT's main campus, the two campuses are well connected via public transportation. Certainly this data could be mined from the web from the relevant Internet site (www.mbta.com), but the gathering of such information would
require a number of laborious steps, a process which itself depends upon some contextual information:

1. entering the URL for the MBTA web site (additional steps if this URL is not known)
2. click the “using the T” link
3. click the “subway” link
4. click the “schedules” link
5. click the “red line” link (additional steps if this choice is not known)
6. go through the vector to determine the current frequency of service (requires time and date)
7. check the special case vector
8. remember this interval in memory!
9-15 repeat the process for the bus schedule
16 determine which transport node is the closest to walk
17 make a comparison of this data as well as “fuzzy” factors like the weather
18 make an informed decision

Note that there are two private services that operate buses along this route – further complicating matters.

Obviously the time required gathering such information represents a greater opportunity cost than the investigation is worth, and the user slips into habit rather than attempts a cumbersome analysis. MapNexus provides utility to the client’s field-based decision-making by delivering information quickly in a given context so that the client may make better short term decisions. The larger issue is to refrain from using the web as a
means to disseminate paper based information (static schedules) and to offer context aware information as transparently as possible (sorted according to the client’s place/time) to permit greater personal productivity and more informed decision-making.

The set of algorithms guiding human behavioral patterns will closely mirror that changing “event landscape” over time. Time is measured in three types of ways: absolute (UTC) time, cyclical time, and time frames. For example, a person who was selecting a lunch option at the Media Lab at 10:32 am on Monday, would as a default, find the following set assumptions: there would be a time frame for the length of the episode as well as a secondary frame to travel back and forth to the lunch. As a default, it would be assumed that one would travel to lunch as a pedestrian or would utilize the proximity of public transportation. Because the context-aware web server would be able to closely coordinate the transportation schedule with the lunch options, a client could better make decisions that were based upon a combination of pedestrian travel and public transport to facilitate decision making.

2.2 Example 2, the Power of Suggestion

One of the central dilemmas of the web in terms of information seeking is the explosive amount of data that is being added constantly. Such a large library of data makes it difficult to find information relevant to a particular context and time frame. MapNexus is designed with attention to the power of suggestion regarding “what is going on” in one’s environment. For example, a Saturday afternoon is generally unscheduled for most students. MapNexus is aware of such time-activity patterns and is therefore in a position to suggest “things to do” that make common sense. Looking at the constraints that can be gathered about a Saturday afternoon in September, it can be represented in a database:

1. A client has a 4-6 hour time frame of activity of a Saturday if the present time is noon.
2. The weather is sunny and warm.
3. The client is physically located on Massachusetts Avenue.

These characteristics could lead to the following hierarchies as to recreational intent:

MapNexus
(1) An activity that will be more appropriate for pleasant weather.
(2) An activity that will require a time-frame of less than 6 hours.
(3) An activity that is open to the public at the present time.
(4) An activity that will be easily reached from MIT.

Thus abstract time periods and possible events can be represented as a series of attributes and algorithms which can be made into a hierarchy of "best fits" between the client's context and an event. In "Negative Expertise," Minsky (1994) argues: "Negative knowledge is involved in many of the forms of thinking that we term 'emotional', notably those involved with humor, shame, fearful, and aesthetic appreciation. This machinery includes a variety of suppressors, critics, and inhibitors, some of which can inhibit not merely actions but entire strategies of thought. Thus, once one begins to look for it, one finds examples of negative knowledge in many activities that we usually see as positive." MapNexus aims at filtering events – a method of suppression or negative intelligence – to allow for a client's preference.

By filtering out information, the client is left with a more manageable set of options. I make the following analogy: the web is like the British Library because it has every book that has been published. MapNexus is similar to a person's reading list that is well indexed, and also MapNexus can easily act as a guide that offers suggestions, based upon both our macro-scaled task (i.e., a free afternoon) or micro-scaled task (i.e., a trip to the nearest drugstore), based on the client's place and time of inquiry.

2.3 Example 3, Cognition of the Spatial Environment

Often we consult online mapping applications to help us get from point A to B. The outputs of such automatic systems are rather brittle and seem verbose to the native. Drawing from the theories of urban form by Kevin Lynch, we know that residents of a city have a spatial index of symbols embedded into memory. Directions to new sites are best conveyed to an individual if they build upon this previously existing spatial knowledge – if the new journey is related to those recollections of previous journeys. (Lynch, 1960). Furthermore an individual
Figure 3 These five visual landmarks are useful to reference when giving directions. A conversational style of directions based upon the client’s context – “It’s just behind the Great Dome” – is a more effective cue to assist a newcomer’s spatial reasoning than verbose sets of instructions – “right, left, right, etc.” A hierarchy of assumptions can be useful to give instructions that would replicate the “common sense” that underpins verbal exchanges.

will have a better chance to memorize new directions and thus is less apt to require a printed map – the new path is presented as a modification of previous experiences in spatial reference. The client of MapNexus, MIT’s academic community, is assumed to have a default set of spatial icons in memory (Figure 3). These urban symbols are assumed to be MapNexus
part of the individual's language (rudimentary spatial knowledge). Each node would itself be a link with a brief description of its location. The goal is to replicate the conversational context between students that is designed to reside in memory:

"walk towards Great Dome" rather than "travel west"
"across from Lobby 7" rather than "go 50M south of the intersection of Vassar and Mass. Ave"

MapNexus aims to replicate the informational exchanges that are found in spoken exchanges. Such a context-based approach has several advantages over present mapping software such as Mapquest that is not context-aware.

First, MapNexus has assumptions about the client's existing knowledge, and MapNexus relates this to the client's present place and time — thus "I'm here" in the above exchange can be modeled to mean far more than just a geometric attribute by including cognitive information. Second, the data is offered to the client in a conversational manner because there is a threshold of assumed spatial knowledge on the part of the user. If this assumption is (Minsky, 1994) proven to be an overestimation, the user can easily remedy the situation by clicking on the icon such as the "Great Dome" for a nested set of meta-spatial information. Furthermore, the goal of MapNexus is to enable the user to enhance spatial cognition in one's memory so that the user does not need to use the computer in the first instant.
3.0 Understanding the Context-Aware Design Approach

"What you need at the outset is a high degree of uncertainty; otherwise it isn’t likely to be an important problem. You start with an incomplete roster of facts, characterized by their ambiguity; often the problem consists of discovering the connections between unrelated pieces of information. You must plan experiments on the basis of probability, even bare possibility, rather than certainty."


3.1 Introducing Context-Aware Computing

This chapter explores the current status of context-aware computing, with particular emphasis on the aspects which are relevant to MapNexus’s ability to model, represent, and interpret context-aware observations (see Figure 5). Currently there are three mechanisms for a platform to determine implicitly the context of a client. The first mechanism is for the client’s computer to send information, based on the client’s past actions in the form of a data packet, to the server which can then return appropriate information to the client. The second mechanism is to determine the client’s location and derive his or her context based upon this entity. The third mechanism is to find out the client’s “intent” based on explicit input into their browser. This chapter examines the leading approaches to determining context based on all of these mechanisms.

Location-based applications are in general far simpler than context-aware applications which are based upon assumptions and human intent in addition to location; the very phrase “location-based” is far more restricting than the more elusive phrase, “context-aware.” Strictly speaking, location-based services (LBS) do not draw from artificial intelligence but are rather more straight forward. Crudely put, LBS are “stupid” in that they just use location as a dimension to filter data while “intelligent” context-aware applications are multi-dimensional and tend to make inferences. While not directly related to MapNexus, there are three larger efforts involving location-based technologies that acted to spurn interest in wireless LBS: the FCC’s 911 mandate for locating mobile phone callers,
the new 3G platform for wireless communications, and the Open GIS Consortium. The FCC has mandated that mobile phone providers give the caller's location to authorities upon dialing 911, the emergency number for the US. Currently about half of all emergency calls are made from mobile phones, and the mandate, due to go into effect in 2005, requires wireless carriers to provide precise location information, within 50 to 100 meters in most cases. (FCC, 2003).

In addition, there is a new 3G mobile platform that is superseding the 2G platform, Global System for Mobile Communications or GSM. 3G offers more integration between wireless phones and the Internet and transmits data using packets-based IP so that a phone will be “always on” and able to receive emails and calls and also to act as a browser with download transfer rates ranging from 14 KB per second up to 2MB per second. Certainly 3G applications will become richer if location can be determined. There is great concern from the public about the issues of privacy regarding 3G/FCC technologies: will the benefits of LBS come at a cost of one’s privacy? MapNexus may appear at first to be a solution to these unanswered questions: MapNexus is anonymous in that the client is never tracked or identified individually; rather the nearest wi-fi node — a link in the IP packet path — acts as link to context-cues.

However, the method to determine location is a side issue. The main point is not designed to be as accurate as the E-911 mandate but rather that MapNexus is more able to utilize a rich “context-aware” approach by connecting interior space with cues to human behavior. MapNexus, unlike mobile phones, works only inside a context-rich environment, interior architecture. While wi-fi systems are almost always found inside buildings, wi-fi is making inroads in portable interiors such as the cabins of Amtrak’s Acela trains and Lufthansa’s planes — all of these spaces are rich in context. Conversely the portability and transmission range of mobile phones creates a context-poor modelling scenario:

What can one assume about a person using a mobile phone that is located 20 miles off Mt. Desert Island (Maine) on an August afternoon?
> not much as person could be a fisherman, sailor, or airplane pilot.

What can one assume about a person using wi-fi connection on an August afternoon in the cabin of a Boston bound Luftansa airplane?
> Quite a lot of assertions can be made: time of arrival to Boston; a need for US immigration
In short, the interior spaces offer a context-rich environment for deploying MapNexus. The accuracy of the assumptions about human behavior in a particular interior reflects the homogeneity of that space: the more specialized an activity within a given physical space permits a great number assertions to be made. Thus the concept of a “context-rich” or a “context-poor” space is less related to the accuracy of a positioning system with space and more related to a predictable land-use patterns on a micro scale.

3.2 Network-Determined Context-Awareness

The first approach to determining context involves the use of a context-aware profile that is derived from the client’s network. An example of a peer-to-peer transfer of context can be examined in Xerox’s ParcTab project, in which PDAs harness the computing power of nearby workstations by communicating with them via infrared signals. Since the ParcTab is “intended to promote casual, spur of the moment, computing” and determines its position from the fixed-location workstations, it is in a position to offer the client location-based information based upon networks’ physical nodes.” (Schilit et al., 1993).

Conversely AvantGo’s recent patent is based upon the notion of “filling” dead spots in a wireless WAN – the server tries to determine which information will soon be required by the client in the short-term and therefore sends back a collection of data which is likely to satisfy the client’s requirements for the short term, thus helping to ameliorate the problem of dead spots or lapses in transmission (Kloba et al., 2003). While AvantGo’s patent is intentionally broad in its scope, the network itself would logically be a method to determine location and to cull information related to the client’s position (i.e., the AP’s IP address). For example, if the client was viewing a web page, it would follow that the client is likely to click on one of the links displayed in the browser. AvantGo’s patent details a method to automatically create and maintain a cache of one’s likely future data required while web
Such an approach requires a method of filtering web content based upon short-term expectations.

3.3 Location-Based Context Awareness

Location can be used as a key cue to determine critical aspects of a client’s context and what is the most useful information needed. For example, the signage presented to a person in an airport varies according to where he or she is: at the terminal, information is sought regarding the departure of all flights, while at the actual gate, only the departure time of the particular flight which leaves from that gate is shown with a greater degree of detail. Only the information relevant to the particular task at hand is revealed. MapNexus aims to utilize exactly this principle in providing information in varying degrees of detail based upon the client’s task.

While MapNexus would be unusual in utilizing a multi-variable, context-aware profile derived from the client’s spatial aggregation unit in a wireless WAN, context-aware technology based on a client’s location is common. Among the most significant location-determining technology is Nibble, a location positioning system for a laptop inside of a wireless WAN; Microsoft’s ‘Location Beacons Methods and Systems’ (LBMS), which provide a framework to determine a client’s location by utilizing radio beaconing technology; and GPS receivers, which determines location by using satellite triangulation. Nibble initially seems very similar to the approach of this thesis because both utilize mechanisms for determining the location of a portable computer client in a wireless WAN based upon the AP (Castro et al., 2001). However, the critical difference is that Nibble operates on the client-side rather than the server-side. By measuring the various strengths of different signals, the Nibble software on a client’s computer can determine its location within approximately 10 feet via triangulation. By contrast, MapNexus simply determines the client’s spatial aggregation unit based on the IP address of the AP transmitter he or she is using; this is far more approximate.
While the more accurate location provided by Nibble may seem initially superior to the framework which MapNexus would use, there are in fact substantial disadvantages to the Nibble approach which would make it unsuitable for MapNexus. First, Nibble operates on the client-side, making it necessary for the client to download (and presumably update) software in order to utilize the technology. In contrast, MapNexus would operate entirely on the server-side and would therefore need no special client-side software. Second, Nibble reports back an accurate location without actually describing the context. By reporting which spatial aggregation unit a client is in, MapNexus provides the server with all the information necessary to determine his or her context because context data could realistically be attached to spatial aggregation units but could not obviously be attached to a nearly unlimited quantity of locations on campus. Nibble is therefore geared towards determining location rather than context.

Microsoft’s LBMS technology aims to connect context with a geographical location. The LBMS method bifurcates the world into a geometric tesselation, a grid akin to the lines of latitude and longitude, and uses this abstract grid in tandem with a more subjective tesselation of geographic data. This approach is similar to the architecture of ESRI’s concept of layered “views.” The Microsoft patent derives a client’s position with a grid; LBMS utilizes radio beaconing as the mechanism for determining the client’s location rather than the AP transmitter of the wireless WAN. In short, the LBMS transmits a “you are here” signal that is designed to act as a widget for a possible context-aware operating system of the future. In this architecture, the operating system itself would act as the middle-ware context engine by giving context to the abstract grid. While this approach is similar to MapNexus, LBMS would require a massive investment in new infrastructure in both transmitters and receivers if a suitable scale of granularity is to be achieved. This US “methods” patent does not provide information as to the expected granularity: an essential element.

The last technology for determining context based on the location of the client is that used by GPS receivers. One example is the MIT ‘Project Voyager’ which was completed in 2001 (Figure 4). ‘Project Voyager’ is a system that attaches a GPS receiver to a PDA, which then enables a client to receive a narrated history of the MIT campus as a function of where
he or she is on the campus. The larger aim of the project was to demonstrate how to build "...a compelling web service for a ubiquitously networked world, where people, places, and things have an Internet presence." (Chan, 2001).

However, one problem emerged with the project during its prototype testing: as GPS only works outdoors, the system would not work inside buildings. Clearly this system would also have little relevance to MapNexus, which is intended to be used exclusively indoors. The second phase of ‘Project Voyager’ is directly relevant to the MapNexus project. Designed for supermarket use, the project involves a customer carrying a PDA in the shopping cart that can scan product bar-codes. The PDA then transmits the data to a server, and much data germane to the item can be offered in the context of the sales environment. MapNexus will operate on a similar basis offering need-to-know information about the client’s immediate situation except that instead of having the client actively send information about an entity to a server, MapNexus will determine the context implicitly by deriving information about the spatial aggregation unit while connecting to the server.

3.4 Intent-Based Context-Awareness

Another approach to finding out a client’s context is to determine his ‘intent’. This can be done by analyzing his or her actions while actually utilizing the computer. This section discusses three different technologies for determining intent: the Mouse Cheese Tool, Just in Time Information Retrieval (JITIR) agents, and Decision Support Systems. The “Mouse Cheese” tool, developed by Mueller and Lockerd at MIT, attempts to determine intent by analyzing movements of the computer mouse (Lockerd, 2001). Mueller and Lockerd discovered that there was a clear relationship between the client’s movement patterns and how the web was being used. By tracking the client’s mouse movements and their timing,
the ‘Mouse Cheese’ tool could discover, for example, whether a client was familiar with a particular web-site or which options they had considered selecting.

The second technology related to intent is the so-called “Just in Time Information Retrieval” agents (JITIR) which use past inputs and patterns of the client to determine context. The aim of the technology is therefore to “… pro actively retrieve and present information based on a person’s local context in an easily accessible yet non-intrusive manner.” (Maes, 2000). JITIR agents are used in applications such as Adobe Photoshop, which is able to predict where a client is likely to want to save a file based on past actions. Strictly speaking, MapNexus will not actually use active JITIR agents. However, MapNexus aims to emulate the aim of JITIR agents – to provide information tailored to a person’s local context – by modeling their likely actions based upon the patterns associated with the spatial aggregation unit they currently occupy.

The third technology used to determine intent is the “Decision Support Systems,” or DSS which aim to aid decision-making by analyzing past behavior-preference models and elucidating what information the client still needs to make a choice. The website Amazon, for example, uses a DSS technology to suggest alternative book purchases for customers based on prior purchasing decisions as well as to allow the client to list books in the order of the “best selling” so that a proper choice can be made based upon other available options (Blanning, 2000). MapNexus draws from these technologies to help the client make a decision in the field about the choices inherent on one’s context regarding a new activity.
4.0 Structuring a Grammar for Human Behavior and Architectural Space

“Philosophy aims at the logical clarification of thoughts. Philosophy is not a body of doctrine but an activity. A philosophical work consists essentially of elucidations. Philosophy does not result in ‘philosophical propositions’, but rather in the clarification of propositions. Without philosophy thoughts are, as it were, cloudy and indistinct: its task is to make them clear and give them sharp boundaries.”


4.1 Micro-Scaled Zoning Ordinance

Consider for example a zoning map (Figure 5) in which certain areas of a city’s ordinance allows for residential, commercial, or industrial land uses. This map is an example of parcel-specific data that has been simplified into a visual representation. MapNexus draws from the idea of the zoning ordinance map: a legal statement of permitted human activities in a given polygon. Zoning maps show a fairly accurate representation of the long term habitation activities. MapNexus seeks to use the concept of a zoning map with its table of “permitted uses” to represent human behaviors in the short term as a function of interior architecture – the land use of buildings and rooms.

In zoning maps, each land-use has a table of permitted uses – activities that people may or may not engage in on this particular premise. Likewise each building has rooms with different functions. MapNexus uses a coherent scheme of visual representation of the MIT campus based on color and also makes logical deductions about a building’s use to create context profiles from the various clues that can be gained from looking at a building’s macro and micro scaled use patterns. There are no buildings at MIT that could be considered truly mixed-use (i.e., to combine chemistry labs and living space). While there are often many activities occurring within the ~50M range of a given wireless AP, rudimentary deductions can be made – tagging a space as recreational, work, or living.
4.2 Wittgenstein’s Language Games

The communication between a client and the web often breaks down because of the failure of the latter to recognize context. Often the problem in mining place/time data is that there are no clear formulas of how variables fit together: just pieces of isolated data such as an address, hours of operation, and events. It takes time and skill to make a quick analysis of the information, especially when information is often displayed in separate sites which force the client to rely upon short term memory or shorthand notes. MapNexus creates a way to impose logic upon a vertical tree of multivariate data that is germane to a particular event. The idea of MapNexus is not so much to define what is possible but rather to define what is not possible. What are the rules of the community in a particular context?

If one views the ecology of information in the Internet as an infinite extension of our own mental capacity for recollection of data, the problem quickly shifts to what Wittgenstein calls “private languages,” the phenomena of thinking and reasoning in language outside the realm of the community of language speakers. The vast data available on the Internet allows MapNexus
the client a large amount of new information; yet one’s ability to understand patterns and connect information to one’s environment is not augmented in tandem. MapNexus thus aims at providing the client information germane to a given spatial/temporal context to allow immediate interpretation with minimum ambiguity. The goal is not to utilize the web to look for obscure data but rather to utilize the web to provide data that could be used in one’s immediate context to help make every-day decisions. Hence, there is an overall view to offer data in logical groupings so that decisions that require comparison or multiple variables can quickly be made.

4.3 Minsky’s “Society of the Mind” approach

MapNexus draws from two approaches to establish the ‘common sense meaning’ of our environment. First, what is common sense? To Minsky (1986), “Common sense is not a simple thing. Instead it is an immense society of hard-earned practical ideas — of multitudes of life-learned rules and exceptions, dispositions and tendencies, balances and checks.” MapNexus aims to be an “intelligent” application and seeks to produce web content that reflects the common sense of a given physical environment at a particular time. One key approach to elucidate “common sense” is the Society of the Mind theory by Minsky that manifests itself as the OpenMind project by his student, Push Singh, at MIT (http://commonsense.media.mit.edu). The OpenMind project collects and organizes large data sets of simple propositions such as, “the sky is blue during the day.” MapNexus could mimic this type of proposition, “Students have more free time on the weekend than during the week.” By drawing from the work of Minsky and Singh, it becomes possible to envision a number of propositions that are related to place and time at the AP spatial granularity. In many ways MapNexus aims not to gather together but to caste away information on the web that is extraneous to the client’s situation.

Once assumptions about the client’s disposition and point of view can be made, MapNexus can then filter event information passing through a net in which the meshes are dependent upon the client’s place and time of inquiry. The net allows all the extraneous events to be cast away and then structures the remaining events into the framework of common sense
possibilities. Thereby MapNexus should help us make short term decisions in a “hour-to-hour”
time frame by offering key bits of data to help one get going: “the big feature of human-level
intelligence is not what it does when it works but what it does when it’s stuck.” (Minksy, 2003).
MapNexus’s goals are not lofty, but more mundane – helping us get to public transport schedules
or suggesting fun things to do on a Friday night. Yet the goals are achievable and represent
the sort of “connectionist” approach to application design by applying Minsky’s theories to the
wireless technologies.

4.4 Natural Language Processing: 12-Dimensions of Verbal Context

The most successful efforts to structure our spoken language is Lenant and Guha’s Cyc
(pronounced “pysche”) markup language that provides a grammar for the unspoken cognitive
context that underpins spoken exchanges. Cyc, which according to its website is “the leading
supplier of formalized common sense,” is funded in part by the Pentagon’s Defense Advanced
Research Projects Agency. Lenant’s work originated from his desire to create a processing
system to enable common sense knowledge related to NLP to be represented. Cycorp
determined that there are roughly 12 separate dimensions to spoken conversations (Figure 6).
By describing the context of a conversation in such a vector, it becomes possible to “virtually
lift” assertions from one context to another. Once a context can be adequately described, it
becomes possible to reason based upon assumptions.

4.5 Determining the Five Dimensions of Context for Architectural Spaces

While Cycorp’s “ontological engineering” has resulted in a sophisticated calculus to
represent spoken exchanges, the problem of the shared context of physical space is more
elementary – there are less dimensions in MapNexus (Figure 6). I have taken as required from
OpenCyc: I have omitted from Lenant’s 12-dimensions several attributes that would not be
CyCorp's 12-Dimensions of Context in a Natural Language Exchange

1. Time temporal projection, relative order, endpoints, granules
2. Type of Time operating with human actors, day vs. night, closing time
3. GeoLocation containment, relative assertions, nearness, everywhere vs. somewhere
4. TypeOfPlace granules of descriptions, cultural uses
5. Culture (Restrictions on Type) actors' subcultures: religion, gender, and regional characteristics
6. Sophistication/Security authority, educational level, permissions
7. Topic subject, preparation, events, primary focus
8. Granularity size of events germane to context, vector of relationships based upon size
9. Modality/Disposition/Epistemology default memory status, expected knowledge
10. Argument-Preference what is the order of items in terms importance?
11. Justification quality of modelling context and the residual utility
12. Let's (Domain Assumptions) a specific case has been set, special conditions

5 Dimensions of Every Wireless WAN's Access Point – the Attributes of Architectural Space

0. Absolute Time
   This metric contains data about the UTC time and makes relationships between intervals; this metric is used by all the other dimensions. Example= September 3, 2003 19:35:12

1. Relative Time
   This metric handles fuzzy units of time – winter, lunch, a museum visit – that are derived from AbsoluteTime and the user's spatial context. Example= Fall Term, rush hour

2. Frame
   The perceived connections between one space and its surrounding environment as a function of both Absolute and RelativeTime as well as the client's mobility. Example= Walking from 77 Mass. Ave. to the ICA in pleasant weather.

3. GeoLocation
   This metric contains information on location – geospatial units and boundaries. Also contains a variety of cognitive representations. Example= lat/lon, "the Great Dome"

4. Disposition
   This metric contains data regarding the intended use of the space as a function of time – the behavioral frame of mind expected within this particular architectural unit. Also this dimension will capture ambiguities in use patterns. Example= research activity, recreational space

5. Hierarchy
   This dimension ranks the relative importance of the above dimensions for a particular information-seeking case. The Hierarchy is akin to a tuning instrument.

Figure 6 The 5-dimensional model for context markup used in architectural modelling is derived from the OpenCyc Markup Schema designed for spoken languages. It is possible to model based on the AP's location and time. What is desired in a context profile is a description akin to the artist's quick sketch of a landscape, taking in both the visual reality of a space (its appearance and purpose) and the psychological character. Again the emphasis is on impressions – trying to represent the essence of a physical place in a few quick marks.

In selecting the above 5-dimensional model to markup interior architectural context propositions, I have created a three tiered schema. First, there is the "atomic" data that has a fixed (1:1) relationship with the AP; the GeoLocation attribute along with the UTC time (AbsoluteTime) form the basis of the context-aware profile. It is this first order "atomic data" that is passed off to a database on the server-side where logic-based rules based rules are executed. These database tables contain attributes that could be bifurcated as being either...
"objective" or "subjective" metrics of client-context. The objective propositions are related to the client’s activity landscape as a function of cyclical time patterns only; I use the phrase, “activity landscape” to refer to the client’s short-term possibilities for a change of activity. Objective data is derived from tables in a many:many relationship and would include such variables as transportation tables of the client’s nearest node of departure as well as data relating to opening hours of MIT’s libraries.

The subjective propositions represent “fuzzy” patterns involving site-specific behavioral expectations and preference associations. These two dimensions (Disposition and Hierarchy) offer hints as to the client’s likely shift from one activity to another over the next few hours. What new data will be required/suggested to assist decision making? Is the client busy or likely to have free time? How can when and where you are reveal clues as to one’s likely short-term intent. These logic-based propositions are based on common sense reasoning; logic-based rules are created to indicate preference patterns as a function of the client’s place and time.
5.0 Capturing Context in Real Time Information Exchange

"As the human-computer interface becomes more pervasive and intimate, it will need to explicitly draw upon cognitive science as a basis for understanding what people are capable of doing. User experience and situation should be integrated into the computer system design process. Situational awareness can be used to reduce the amount of explicit input a person is required to give a computer. Contextual information of what and where the user task is, what the user knows, and what the system capabilities are, can greatly simplify the user scenario. Such use of contextual models in computers can also reduce the teaching needed for the user to accomplish tasks."


5.1 Presenting the Role of Middle-Ware

In human-computer interaction (HCI) on the web, the computer is rarely tasked with elucidating "cloudy and indistinct" inquiries because open-ended questions depend upon common sense reasoning. Without cues to the client's context, an unspecific request on the internet results in too much information which is laborious and confusing to analyze. The technology and structure of the web — viewed as a living ecology of information — controls our ability to harvest useful data related to a particular time and place. The laws of economics are useful to consider in HCI, for the large amount of information available online makes the opportunity costs too high to search the internet for anything that is of a "cloudy and indistinct" nature. Each click and keystroke is a spent currency of sorts — with MapNexus I am trying to lower the cost barrier to quickly find useful local information. The role of the middle-ware is to add common sense reasoning to a given intranet.

Considering the following inquiry: "What fun things are going on this Friday night?" The answer requires some common-sense reasoning ability. A student could check a number of websites (yahoo movies, MIT arts calender, etc); each will require the client enter data relating to the date of inquiry as well as the venue. Since there is no clear method of implicitly modeling context on a broad scale across many web sites at once, narrowing down such a question by location and personal preferences is impossible. The client quickly learns to avoid making MapNexus
such inquiries in the first place; the task of searching for the right data can be quite irksome. Yet such a broad and unspecific query could indeed be answered by an intranet server that was designed to handle common-sense reasoning about the client’s context and the surrounding environment.

The barrier to the flow of site-specific information from the client's perspective is not connected with the speed of the network, but rather the time it takes for the user to cognitively process the large quantity of data and then to sift until only the desired data remains. This threshold of effort is high enough to render some data-seeking activities worthless as its capture has a higher opportunity cost than its utility. A “middle-ware bucket” of context cues can be used both to save the client time and to provide utility by reducing the need for explicit input (clicking and typing) in repetitive information-seeking tasks that one finds in one’s day-to-day life, a pre-determined urban environment in which the client knows many common sense rules.

The overall question regarding the MapNexus server becomes, “What role can a network play to assist in detecting site-specific behavioral patterns inside building? How can a network infrastructure become a tool to add commonsense that relates to architecture?” The bifurcation between the client and the server – the front-end and the back end – appears iconic in computer science. MapNexus relies on harnessing the middle ground -- the physical transmission media of air and wires. MapNexus detects the context of the client by looking carefully at the location of the nodes of physical transmission (figure 7). The AP lies in close proximity to the client in a wireless WAN as the radio frequency transmission range is roughly 50M; there is a definite pattern of geographic uniformity in the granularity of the AP cells.

In the MapNexus framework, the Middle-Ware (or “Context-Engine”) produces within a fraction of a second a computer generated representation — a snapshot of the client’s present context — by combining objective data (place and time attributes) with subjective data (behavioral assumptions, preference patterns, and pre-existing cognitive knowledge) to produce common-sense rules a “context-aware profile” that is designed to provide a grammar for MapNexus server to interpret. The intranet server is specially designed for information-seeking regarding the clients’ environmental character within a specific WAN, and with certain modifications can be made to produce web content germane to the client’s situational decision-making.
Capturing the Context of Architectural Space

http://mapnexus.mit.edu

**Step 1:**
The client connects to a wireless AP on campus and pulls up the MapNexus URL.

**Step 2:**
The request is received by the server, and the AP node's unique IP address is detected and passed along to a server-side RDBMS.

**Step 3:**
The RDBMS makes a 1:1 match with using the AP's IP address as a primary key to index an atomic table of fixed data.

**Step 4:**
The RDBMS uses the atomic data to link to several tables containing rules about environmental variables as a function of time.

**Step 5:**
The RDBMS creates a "context-profile" that contains many assertions; this profile is used to create a context-aware web site.

*Figure 7* A specific architectural space's characteristics can be woven together with the internet in a wireless WAN utilizing a middle-ware RDBMS to aggregate data to gain cues as to the client's short-term activity frame.
5.2 AP Polygon as a New Unit of Spatial Aggregation

The engineering feat of the 802.11 wireless protocol has a side effect of creating new geographical units of space – the AP cell or geographic area surrounding the AP – that can be modelled in a database as an entity with an array of attributes to establish context-aware cues. This entity-attribute model and geometric tesselation recall the organizational structure of the Zoning Improvement Code (a ZIP code was conceived as a routing shorthand for parcel delivery) which became the de facto unit of demographic modelling in the USA. MapNexus will place the client within a particular building on MIT’s campus — a unique parcel of real estate with a particular function (Figure 8). MapNexus follows a hybrid raster model in that it is not a simple grid overlay (as many places on campus have no wi-fi access), but rather a clustering of APs that reflects the architectural landscape and function of the buildings. While there may well be some ambiguity as to exactly which AP the user is assigned (most wi-fi systems constantly compare signals of base stations) the user in most instances assigned to the nearest base station. Exceptions to the rule occur when there is substantial physical obstacles, and as a fail safe the user is permitted to set one’s AP manually (by location) to prevent glitches that for example can occur in corner offices where the many windows allow a stronger signal from a neighboring building to reach the user’s laptop as opposed to a signal that is may well be closer to the user but degraded due to physical obstructions such as walls. The overall point is to allow the user to be placed within the urban fabric which can be described.

MapNexus is only designed to provide web content within an urban fabric with a predetermined wireless WAN — MapNexus cannot make assertions about clients who make inquiries without preexisting data that correlates to an architectural space. For a MapNexus type system to exist, there first has to be a “context-rich” geography of the physical intranet — the wires and APs — within an urban fabric. Moreover, this urban fabric has to be modelled in a pre-existing database of site-specific “common sense” propositions and data sets (such as schedules). Today one’s user experience of the web is completely estranged from one’s physical context. Wi-fi could provide a means joining these two separate realms – the client’s physical surroundings and the client’s web content — to gain a more conversational approach that is based on Minsky’s notion of “common sense” knowledge.
Figure 8 This AP is located in a chemistry laboratory (rm 157) and would be modelled exactly the same way as the AP in Rm 110 as the most logical tesselation would be to group this cluster of APs by land-use pattern; Likewise the dorm to the south (East Campus) would be clustered at the building level (having several APs). The granularity is based upon differences in land-use patterns in architectural spaces.
5.3 Harnessing the Latent Data in the Client’s “Get” Command

When a client makes a request for an intranet site, an upstream HTTP “get” command can be parsed to determine which AP node is crossed as the packets move upstream in the first hop from the client’s laptop to the physical media or transmission cables. At MIT, the tradition is to name the Internet protocol (IP) address of the AP for its physical location which makes it very easy to create an index of the ~450 APs in the Athena system (Figure 8). The atomic unit for the context-gathering is the AP which is in MIT’s WAN is conveniently given an IP address that is synonymous with the AP’s physical location. There are several applications that are based on the UNIX “traceroute” command that are specifically designed to identify the route of a specific IP transmission from the server side (Figure 9). Therefore it is within the network administrator’s capability to make available the AP’s IP address to the MapNexus server when an incoming HTTP request occurs. By identifying the AP, the client’s position is determined implicitly.

A question that begs to be answered: “If the location of the client is fairly easy to establish, why are there not already context-aware websites?” First, there is often little correlation between the physical location of the client’s nearest network node (a fixed physical piece of hardware) and client’s actual location in a sub-net or local area network (LAN). Historically, MIT has been organized into LANs that are structured by departmental administration — thus it is possible to have a LAN that is unusually shaped. Second, a large amount of Internet traffic is handled by dial-up connections (ISP), and this arrangement offers few cues to be placed in the upstream IP/TCP packet envelope to establish a consistent spacial relationship between the client’s location and the nearest network node.
Figure 9 These screen shots show network administration applications that are designed to provide data regarding wifi IP packets: routing, MAC addresses, and IP addresses. A `traceroute` command displays the routing of a packet from my parent's house in North Carolina (see left panels) to the wireless AP in the basement of MIT's building "E2." The Commview software "sniffs" wireless IP packets.
5.4 Representing Context in a Relational Database

During this project’s inception, I wrote the MapNexus transportation schedules in javascript in the HTML page, and later I tried to develop a parent-child model for all the variables. However, these approaches would not be feasible considering the scale of MapNexus. The relational database model has many advantageous to other approaches if ease of updating the variable data is essential to the project’s goals. A well structured relational database (entity-attribue model) is essential to connect the client’s context with related events. The goal of the SQL queries is to compute on the fly a shortlist of the events that might be possible for the client to engage; the relative preference that a client will have will (via additional set of algorithms) help narrow down this list by ranking and sorting. The first SQL query creates a table of events by selecting only those events that are (1) relatively close-by, and (2) are valid for the present time frame. Since the amount of events is likely to exceed the limited set desired by the client, second order queries could be run to filter the events further by ranking these events (see Chapter 6).

The AP’s static IP address is used as a foreign key in the AccessPoint Table \(C1\) which contains a number of other attributes of the client’s space: building and room number, MAC address of the AP device, and a primary key being assigned via a serial number \(AP\_ID\). This table is linked to several others in a many to one relationship (see Figure 11). Table \(C2\) places the building within a larger polygon of space \(zone\_id\) and also gives a geometric representation (Cartesian Coordinates) in the form of the Massachusetts State Plane Grid to permit SQL queries using Euclidean Geometry to compute distances.

In a relatively small campus like MIT, events can be considered to be either “on or “off” campus but having the coordinates will allow further development at a later date (in a more complex geographic WAN). Table \(C3\), Bldg\_Use, introduces cyclical time patterns to the location — again permitting SQL queries as a function of the client preference that would be expected within a certain building at various times of the day. Most important is the profile\_id in Table \(C4\) that refers to a preference profile vector as a function of time for the building; this ~30
site-specific behavioral attributes

Step 1:
Attributes about every AP are stored in a server-side RDBMS before the client connects.

server/ AP "sniffer"

Step 2:
The client connects to the MapNexus website and the AP of origin is sensed implicitly.

content engine/ databases

Step 3:
By modeling the client and event context profiles in a relational database, SQL queries and related algorithms can be performed at the time of inquiry to offer commonsense info about one's immediate environment and likely preferences in the next few hours.

web page

Step 4:
The RDBMS produces content to fill the web page based upon real-time ranking of the "best matches" for the events for each AP zone.

Figure 10 The MapNexus web page is based upon common sense knowledge about both the client's behavioral patterns and shifting environmental possibilities.

MapNexus
variable array is crucial to the match making process and will be discussed at length in Chapter 6. Finally, table C5 connects the client’s building to the nearest node of transport.

Transportation schedules — the ability to get real time data based upon a client’s location — is a data intensive operation. Tables T1-T5 act together to model the schedule, location, mode, and operating times of the regular transportation routes such as the subway or shuttle services that allow to client to quickly move to a new physical area and thus have a wider range of possible activities. Each route has stops or “nodes” that can be places within a given zone_id via the coordinates. The way in which the schedules appear in the web pages’s GUI shall be discussed in Chapter 6.

Events (like the AP) are modeled to denote the proximity, time, and preference profile of the activity so that matches can be made via SQL queries. So that cyclical events can be described once while accommodating flexibility and repeated schedules, the scheduling data is contained in a separate table. Note that the profile_id field contains a serial number that represents the ~30 variable vector in a separate table (C6) just like that assigned to the client AP via a static array of variables. Both Client and Event data includes a profile_id field to allow for comparisons to be made. Table E2 allows for complex schedules to be handled for each event and for changes to made with ease.
### CLIENT CONTEXT

<table>
<thead>
<tr>
<th>AccessPoint C1</th>
<th>PK</th>
<th>AP_ID</th>
<th>PK</th>
<th>ap_ip</th>
<th>18.89.1.101</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPAddress</td>
<td></td>
<td>ap1.5-321.mit.edu</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MacAddress</td>
<td></td>
<td>A1-49-00-00-00-FF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Room</td>
<td>PK</td>
<td>533</td>
<td>PK</td>
<td>room_id</td>
<td>PK</td>
</tr>
<tr>
<td>Bidg_ID</td>
<td>PK</td>
<td>9</td>
<td>PK</td>
<td>bidg_id</td>
<td>PK</td>
</tr>
<tr>
<td>AP_Depth</td>
<td>PK</td>
<td>11</td>
<td>PK</td>
<td>ap_depth</td>
<td>PK</td>
</tr>
</tbody>
</table>

### BUILDING USE CONTEXT

<table>
<thead>
<tr>
<th>Building C2</th>
<th>PK</th>
<th>bidg_id</th>
<th>PK</th>
<th>zone_id</th>
<th>C3</th>
</tr>
</thead>
<tbody>
<tr>
<td>stateplane_x</td>
<td></td>
<td>233000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stateplane_y</td>
<td></td>
<td>901000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### EVENT CONTEXT

<table>
<thead>
<tr>
<th>Event E1</th>
<th>PK</th>
<th>Event_ID</th>
<th>PK</th>
<th>Event_scheud E2</th>
<th>PK</th>
<th>Event_Prefer E3</th>
<th>PK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event desc</td>
<td>PK</td>
<td>e-planning seminar</td>
<td>PK</td>
<td>term fail</td>
<td>PK</td>
<td>a1</td>
<td>PK</td>
</tr>
<tr>
<td>event_cat</td>
<td>PK</td>
<td>Seminar</td>
<td>PK</td>
<td>week all</td>
<td>PK</td>
<td>a2</td>
<td>PK</td>
</tr>
<tr>
<td>sponsor MIT</td>
<td>PK</td>
<td>MIT</td>
<td>PK</td>
<td>day weekday</td>
<td>PK</td>
<td>a3</td>
<td>PK</td>
</tr>
<tr>
<td>start_date</td>
<td>PK</td>
<td>9/1/03</td>
<td>PK</td>
<td>start_time</td>
<td>PK</td>
<td>a4</td>
<td>PK</td>
</tr>
<tr>
<td>end_date</td>
<td>PK</td>
<td>12/12/03</td>
<td>PK</td>
<td>enc_time</td>
<td>PK</td>
<td>a5</td>
<td>PK</td>
</tr>
<tr>
<td>language</td>
<td>PK</td>
<td>MIT-only</td>
<td>PK</td>
<td>e_detail</td>
<td>PK</td>
<td>a5</td>
<td>PK</td>
</tr>
<tr>
<td>on-campus</td>
<td>PK</td>
<td>yes</td>
<td>PK</td>
<td>e_constraint</td>
<td>PK</td>
<td>a5</td>
<td>PK</td>
</tr>
<tr>
<td>o_place</td>
<td>PK</td>
<td>901133</td>
<td>PK</td>
<td>e_bldg</td>
<td>PK</td>
<td>9</td>
<td>PK</td>
</tr>
<tr>
<td>e_room</td>
<td>PK</td>
<td>556</td>
<td>PK</td>
<td>e_bldg</td>
<td>PK</td>
<td>9</td>
<td>PK</td>
</tr>
<tr>
<td>event_url</td>
<td>PK</td>
<td>/eseminar.html</td>
<td>PK</td>
<td>profile_id</td>
<td>PK</td>
<td>321</td>
<td>PK</td>
</tr>
</tbody>
</table>

### TRANSPORTATION CONTEXT

<table>
<thead>
<tr>
<th>NodeData T1</th>
<th>PK</th>
<th>node_ID</th>
<th>PK</th>
<th>node_name</th>
<th>PK</th>
<th>node_x</th>
<th>PK</th>
<th>node_y</th>
<th>PK</th>
</tr>
</thead>
<tbody>
<tr>
<td>node_name</td>
<td>PK</td>
<td>Kendall-1</td>
<td>PK</td>
<td>233250</td>
<td>PK</td>
<td>901236</td>
<td>PK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bus_stop</td>
<td>PK</td>
<td>yes</td>
<td>PK</td>
<td></td>
<td>PK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>taxi_stop</td>
<td>PK</td>
<td>no</td>
<td>PK</td>
<td></td>
<td>PK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>subway_stop</td>
<td>PK</td>
<td>no</td>
<td>PK</td>
<td></td>
<td>PK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>shuttle_stop</td>
<td>PK</td>
<td>yes</td>
<td>PK</td>
<td></td>
<td>PK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Sample QUERY to find events relevant for DUSP student in building 9 at 3 PM on Friday, December 12, 2003

Create View events_of_interest

as Select e.eventID, e.event_desc, e.start_time, e.end_time, e.bidg, e.room, en.map.id, event_match, time, value(event_match, u.term, u.week, u.day), "current_date-time", b.bidg_id) event_value

from Event e, Event_Schedule s, Event_Type t, Building eb, map em, AccessPoint a, Building b, Bidg_use u

where e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

and e.event_desc = e.event_desc and e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

and e.event_desc = e.event_desc and e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

and e.event_desc = e.event_desc and e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

and e.event_desc = e.event_desc and e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

and e.event_desc = e.event_desc and e.start_time = e.start_time and e.end_time = e.end_time and "current_date-time" = e.start_time + e.end_time + "current_date" + "bidg_id"

#### Create View event_proximate_places

as Select e.bidg_id, "walk" node, SQRT((e.place_x - b.stateplane_x)^2 + (e.place_y - b.stateplane_y)^2)/100 time

from Event e, AccessPoint a, Building b

where e.bidg_id = current_ap and b.bidg_id = bidg_id

and e.bidg_id = e.bidg_id and (e.place_x - b.stateplane_x)^2 + (e.place_y - b.stateplane_y)^2 <= 500 SQLT

union

Select e.bidg_id, d.node, d.time

from Event e, AccessPoint a, Building b, Node00 d, BidgNode bn, BidgNode en, VehSchedule v

where e.bidg_id = current_ap and e.bidg_id = bidg_id and e.bidg_id = current_ap and e.bidg_id = current_ap and e.bidg_id = e.bidg_id

and e.node_id = en.node_id and e.node_id = bn.node_id and e.node_id = en.node_id and d.node_id = v.node_id

and en.node_id = v.node_id and e.node_id = en.node_id and (e.place_x - b.stateplane_x)^2 + (e.place_y - b.stateplane_y)^2 <= 500

and (en.time + d.time + bn.time + v.time <= 60)

get events with event-match info and proximity information

tables to be joined (building appears twice, for event and for client locatior match current time intervals)

for simplicity use travel time = 15 minutes

e_min_time is minimum time for event to be valuable

join the tables

join ap table to client building too

filter event types

walkable events

tables to be joined

join the tables

current AP within 500 meters

non-walkable events

tables to be joined (BidgNode appears twice, for event and for client locatior)

join the tables

look at the relevant bus schedule and assume random arrival within headway

current AP not within 500 meters

event node within 60 minutes
5.5 Scale of the SQL Operations

The RDBMS in Figure 11 illustrates a method to perform SQL queries to create a shortlist of events that are within the event landscape of a client in a given AP polygon. The size and speed of the SQL operations are important since the system is engineered to provide seemingly instantaneous output to the end user. Before discussing the size and scope of the SQL server operations, it is important to note that it would not be necessary to complete this entire process each time a client requests the MapNexus URL; one could well imagine that in a lecture class with 100 students arriving at 9:00 AM (all checking MapNexus upon arrival) thus creating a spike in terms or repetitive enquires. Hence a method to cache the output of a given query for a limited amount of time will greatly speed the system.

A key question is: “What is the size and scope of the RDBMS and its SQL operations?”

Currently there are approximately 500 access points on campus. We anticipate differentiating the time aspect of a client context by intervals as small as 5 minutes. Since there are approximately 100,000 5-minute intervals in a year, an exhaustive listing of all time-place contexts would yield 50 million cases for the MIT campus. Matching client and event context using a table of that size would be impractical. However, for any particular location, there are many cyclical patterns to the context. Most weekdays are the same, weekly patterns tend to be similar within a semester, and most daily patterns exhibit a morning, noon, afternoon, evening, and night pattern. Instead of distinguishing every 5-minute interval, the relational tables store the start and end time of each context period. An academic building might then have only 4-6 different contexts during a typical mid-semester weekday. With this in mind, the number of different client contexts that we need to distinguish might be:

\[
\text{(the amount APs in a building)} \times \text{(number of building)} \times \text{(terms)} \times \text{(weeks)} \times \text{(days)} \times \text{(time-of day)} = \text{total number of rows}
\]

\[
5 \times 100 \times 5 \times 4 \times 3 \times 6 = 180,000
\]

Whereby we distinguish 5 semesters (summer, fall, independent activities period, spring, and vacation); up to 4 different types of weeks within any semester; 3 different types of days (weekend, weekday, holiday); and 6 different times-of-day. Some further pruning would likely be possible, but the anticipated SQL queries would be manageable on the server with modern RDBMS software and a 180,000 row table of client contexts.

We estimate that there might be up to 1,500 event that are available for consideration.
on a given day. For the system to be practical for users, the context matching would have to reduce this number down to the most relevant dozen or two. The basic joins described in Figure 11 might eliminate 50-80% of the events due to the event’s hours of operation but that would still leave 300+ events. Once this “short list” of events is created, the question becomes, “What events are the most likely to be of interest to individuals as a function of preference patterns? How shall these events be displayed to the end user to maximize utility?” The ‘preference profile’ assigned to each client and event context would then be used to rank order these remaining events. The use of these preference profiles is explained in the next chapter along with a user interface design that would allow the system to be practical and useful in terms of the client’s cognition of what events are available.
6.0 Designing a Context-Aware Web Server

The task of the logothete, or the founder of language, is an endless cutting up of the text: the primary operation is to ‘grab’ the cloth in order then to pull on it (to pull it off). We must then therefore in some measure make a distinction between deciphering and cutting up. Deciphering refers to a pregnant depth, to an area of relationships, to a distribution... and of quite another order of importance is cutting up – and putting into a system – this text that is no longer a reductive reading, but an exalting, integrating, and restorative reading.”

Roland Barthes, 1971, Sade, Fourier, Loyola.

6.1 Processing Context to Event Matches Profiles

Returning to the illustrative examples in Chapter 2, MapNexus provides context aware data in two user scenarios. The first scenario involves a person who is trying to quickly find the schedule of the public transport options from MIT to Harvard Square for immediate use (Figure 12). The second scenario involves an unscheduled period of time on a Saturday; a student has free time and is curious as to what events are of interest in terms of taking a break from academic work (Figure 13). Both questions require some “common sense.” Representing common-sense reasoning into MapNexus is a complex undertaking.

In matching context with events, MapNexus has relied upon a logic-based approach to filter the event landscape related to the client’s particular AP profile via a RDBMS. After the shortlist of events has been created by the SQL operation, the next step involves creating an ordered list of “best fits” between the client’s context and the events within the client’s landscape of possibilities. This further refinement lies at the core of the middleware operations: algorithms to select, rank, and group events that are most applicable to a given context. The overall process from start to finish is listed as follows:

1. Represent all events and APs and events as having a “context profile” in a RDBMS.
2. Perform SQL queries to make a short list of events germane to the client’s place and time.
3. Refine this list further by applying an algorithm to create a hierarchy for the best matches.
4. Fill the MapNexus home page with site specific information.
GUI homepage ./index.html

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>now...</td>
<td>Open House Center for Bits &amp; Atoms</td>
<td>Demo Kitchen of the Future</td>
<td>Visual Arts Lottery Open at List</td>
</tr>
<tr>
<td>soon...</td>
<td>Lecture Ishii’s Latest Work</td>
<td>Student Council Candidates Speeches</td>
<td>Public Service Doctors w/o Borders</td>
</tr>
<tr>
<td>ahead...</td>
<td>Exhibbit History of Calculators</td>
<td>Film LSC presents “Kes”</td>
<td>MIT Sports Championship Ice Hockey</td>
</tr>
</tbody>
</table>

GUI linked page ./transportation.html

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>transportation kendall square</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>red line</td>
<td>outbound 3:37</td>
<td>3:45</td>
<td>4:53</td>
</tr>
<tr>
<td></td>
<td>inbound 3:39</td>
<td>3:47</td>
<td>4:55</td>
</tr>
<tr>
<td>ct#1</td>
<td>outbound 3:56</td>
<td>4:14</td>
<td>4:34</td>
</tr>
<tr>
<td></td>
<td>inbound 3:36</td>
<td>3:49</td>
<td>4:15</td>
</tr>
<tr>
<td>86</td>
<td>outbound 3:36</td>
<td>4:00</td>
<td>4:34</td>
</tr>
<tr>
<td></td>
<td>inbound 3:39</td>
<td>3:44</td>
<td>4:16</td>
</tr>
</tbody>
</table>

Figure 12 The Transportation links are found by a link in the bottom menu of the home page.

MapNexus
User Scenario #1 (previous page)

The client is new to MIT and unfamiliar with the building naming system; the client is using a laptop computer in the Media Lab (Building E-15) and seeks information about where and when one can use public transportation to travel to Harvard Square.

Since the user is making an enquiry to the MapNexus server from an AP in E-15, assumptions are made based on the time (late afternoon) and the place (an academic research facility specializing in multi-media computing). Events are listed that relate to either present context or the expected change in context as Friday night arrives (hence the "ahead" grouping of events are primarily social). Throughout MapNexus, "Transportation" and "Eating" are located in the bottom menu to aid in navigation -- these events are special cases that are never null sets. In this scenario, the client is quickly provided with the name of the nearest transportation node (Kendall Square) and the range of options with various departure times by clicking the link to the ./transportation.html page. If the client wishes to see the schedule of a less popular routes, the client clicks the "more" link from the ./transportation.html page.

Notice how the options for events under the "now" category (those starting within 15 minutes or have already begun) are generally related to the function of the building -- 3/4 of the choices are directly related to the Media Lab. This set of events is in contrast with the "ahead" category. Since the Media Lab is assumed to keep regular business hours, the weighting mechanism has recognized that the client will likely engage in a change of context considering Friday night is rapidly approaching -- a time in which people usually engage in social activities. Therefore the "ahead" category offers events assuming a switch in behavioral context -- social and recreational events.
GUI homepage ./index.html

now... Social E66 Residents’ Brunch
Music Lewis Library offers free CDs
Recreation MITOC open house
Business UROP leadership program

soon... Film Harvard Film Archive premiere
Music Digital Music Recital
Comedy 24 hours of Standup
Literature Poetry Workshop

ahead... Film MFA shows Scandinavian Films
Exhibit History of Calculators
MIT Sports Championship Ice Hockey
Social Bashdown Party

GUI homepage ./mfafilms23.html

film MFA shows Scandinavian Films

11:30 this cinematic jewel deftly mixes stories of lives in a small town as they unfold simultaneously over the course 1 hour. $8

> Take Bus CT#2 from MIT to the MFA.

Figure 13 An off-campus event appears during a Saturday morning when the student is less busy.
User Scenario #2 (previous page)

The client goes to the MapNexus site after waking up in the East Campus undergraduate dormitory. The client has some free time because it is a Saturday morning and wishes to take a break and see a cultural event. The client is seeking a suggestion and has no preconceived notion of exactly what event would be worth attending.

In this case, a client is located within 100M of the Media Lab -- on the other side of Ames street in an undergraduate dormitory. Since the inquiry is made on a Saturday morning, it is assumed that the client will have an unscheduled day and will thus have a likelihood of leaving campus for a cultural excursion or, if staying on campus, will likely engage in an artistic or recreational activity. The first set of options include a free brunch being held in the dorm and an event relating to a student group: the MIT Outing Club. The second group of events that start in the afternoon include several art events: a film, a recital, and a literature workshop. Finally the “ahead” category features an event that looks promising -- a Scandinavian Film at the Museum of Fine Arts beginning at 11:30.

An obvious question, “What if the student decides to study all day; why are there no links that have work-related events?” The reason there are no events relating to the academic environment displayed on Saturday morning is that there is no specific “event” of note. Naturally the student can go to the library and study all day, but this would be done alone and there would be no clear start or end time (the student could study anywhere). In other words, a event need not be listed to be a real possibility. Rather the events are listed because the client requires some specific data in order to make a choice. Therefore many “obvious” choices like studying or sleeping are not listed at all.
6.2 Client/Event Preference Vectors

Now that client experience has been illustrated, the server-side algorithms that refine the SQL queries must be elucidated. As discussed in Chapter 5, the event data can be filtered to remove those events not related to the AP’s place and time. At this point the I will focus on how the event and client context can be matched using the data. The goal of step #3 is to rank possible upon the client’s preferences. The server is able to make refinements to the match making process akin to the subjective dimensions as in OpenCyc. While the objective selection detailed in the SQL queries is relatively, the ability to rank events based upon subjective metrics is much more complex every match between a specific event and a given context is relative to the other possible events. The following expression is used to establish a value for the event’s relative fit:

\[
\sum_{\text{ap_id: event_id}} = ([a_1, a_2, a_3, a_4, a_5, a_n] \times [b_1, b_2, b_3, b_4, b_5, b_n])
\]

The value, \( \sum_{\text{ap_id: event_id}} \), is the sum of the product of the two vectors— the first representing the client’s context profile and the second representing the event’s profile. The \( \sum_{\text{ap_id: event_id}} \) value is a relative value -- the score in itself means little. A higher value indicates a better fit between an particular event and the client’s context.

The variable “a” and “b” store the assigned weight (metric for like) in increments. This array of values need not be binaries but should be a range (i.e., 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10) of integers which a low value indicates a low likelihood and a higher value indicates a greater likelihood for a match along any single dimension. This approach allows great flexibility in programming as the system can be adjusted by either changing the intent vector for either the client or event at a later time; it is important to remember that flexibility is crucial as one ventures into the cloudy and indistinct realm of predicting intent. Each vector would consist of an array of approximately 30 variables representing the client/event profile along a variety of metrics. The assignment of variables would remain constant for each event but shift as a function of time for MapNexus.
each AP context mirroring the fact that our behavioral patterns shift as a function of time. The profile of each event/client context would be stored in the RDBMS tables and linked using the primary keys (Event_ID and AP_ID plus the relative and absolute time data) from the database.

The events that are retained for each client context are sorted from greatest to least via the $\sum_{ap:event_id}$ value. The match making and ranking queries are then run three times with the time set to “now,” +30 minutes, and +90 minutes; no serial event is to be picked twice.

<table>
<thead>
<tr>
<th>Occurs in</th>
<th>Type I “now”</th>
<th>Type II “soon”</th>
<th>Type III “ahead”</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-30 minutes</td>
<td>0-30 minutes</td>
<td>30-90 minutes</td>
<td>&gt; 90 minutes</td>
</tr>
<tr>
<td>x4 1st choices</td>
<td>x4 1st choices</td>
<td>x4 1st choices</td>
<td></td>
</tr>
<tr>
<td>x16 2nd choices</td>
<td>x16 2nd choices</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

As far as structuring the type of output: type I events are limited to the immediate time frame. The type II choices assume a wider ken and would include any event that has a start time in the next few hours. Finally the type III group of events represents an “up next” suggestion that will help the client to plan ahead with a time frame extending into the next day. The system could be further refined by using different client context weights for the three time frames. Additional query conditions could also ensure that any event appears only once in the earliest possible of the three categories.

From the client’s perspective, the MapNexus site will have a set uniform resource locator (URL) which the client enters into a browser; the web site’s GUI will be akin to both a search engine and a local event’s calender. A database driven web page is created on the fly using some combination of a dynamic content mechanism (PHP, XML, JDBC, and the like) to generate output based on SQL queries that match and rank events based upon the user’s context-profile (derived from the AP). Nothing more than javascript and XML capabilities are required at the client side. The main point is that the client is not required to have any additional software.

The client can clearly see the AP he is associated with (or change this variable via the toggle in the home page). By clicking on an event, the user is offered the details and in some cases other lateral links. The three types of events (“next, soon, or plan”) are links themselves that reveal the “B” list of 16 events. Thus within three clicks, the user would have access to about 120 separate events (assuming some lateral links).
6.3 Illustrations of Matching Event to Context

The system's logic can be illustrated using an abridged version of the client/event vectors; these vectors represent the preference associations along a certain dimension. The example below illustrates this idea of showing (1) a close match between event and client and (2) a poor match between event and client context. The actual system would have a few dozen dimensions -- one for each academic departments and other more subjective dimensions drawn from the OpenCyc language (discussed in Chapter 4):

<table>
<thead>
<tr>
<th>Event: E66, a residential dorm</th>
<th>Value: Digital Narrative Lecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>academic -- a connection with computer science</td>
<td>[2 x 9] = 18</td>
</tr>
<tr>
<td>break -- a need for a change of activity</td>
<td>[5 x 1] = 5</td>
</tr>
<tr>
<td>excursion -- the viability of a journey off-campus</td>
<td>[9 x 0] = 0</td>
</tr>
<tr>
<td>residential -- a connection with living space</td>
<td>[8 x 0] = 0</td>
</tr>
<tr>
<td>social -- a venue for meeting others casually</td>
<td>[5 x 3] = 15</td>
</tr>
<tr>
<td><strong>Total</strong>: 38 low</td>
<td></td>
</tr>
</tbody>
</table>

From the above scores (10 being a high preference and 0 being a low preference) the Digital Narrative Course -- being taught on a Saturday morning! -- would not fit with the client’s context. The client is more inclined to take a break off-campus and do something social and less academic. On the other hand, if the event was held during the week or if the location were a computer science facility, it would receive a much higher score as the client’s preference profile shifts as a function of cyclical time and place.

Similar to OpenCyc, the metrics are able to store variables that represent diverse dimensions in context. Note in the above example subtle difference between “break” and
“excursion.” The first represents a shift to a new activity without respect to time or space -- it may be as simple as stretching one’s arms. “Excursions” is a metric that measures the client’s desire to leave campus and the event’s physical location away from campus. Note also that the “social” metric is labeled “casual” so as to differentiate between socializing with one’s peers and socializing romantically which is often a function of time. In total, the vectors provide an essential framework for placing common sense reasoning into a web server that is operating in real time. Furthermore, the framework allows for events that do not have clear start and end times -- or those that do not have a spacial component -- to be compared with other events.

6.4 Site Maintenance and Correction

This system will not run itself and will require constant maintenance – pruning – to allow for the right growth to occur in the ecology of information. Changes in building use, new APs, and shifting polygons will all need to be addressed as time passes. User feedback is an important part of the process to correct poorly executed assumptions, and an web-based form could be employed to collect debugging info and send to the remote server for the web master’s review.

The MapNexus system could be designed to allow any authorized Athena account holder (MIT authorized affiliate) to add events via a web form to the server database so that new events can be uploaded into the system by the community. Part of the process of adding new events would include the assigning of weights to the dimensions from a limited amount “points” to ensure that no event receives “10” scores across all the dimensions. Obviously, the weighting is a fairly complex and would be best achieved be designed a well structured web form using that gathers information by asking a series of questions akin to the process of listing an item on Ebay where the categorization process via a web form has been well-developed and allows for cross-listings in which one item can appear in multiple lists. Unlike Ebay, MapNexus will also ask the user for data relating to more subjective dimensions -- for example the event’s ability to act as “break” or the event’s
social character.

Regarding the actual visual layout of the MapNexus server, I wanted to use a set of tables and graphics to help create a sense of structure for the home page. The home page would consist of a number of links to various data -- thus the MapNexus website has only two layers. I designed a test version of an XML pages for this project in 2002, but later abandoned the design as it utilized too many visual maps, and was not focused on event-context. A second layer of the MapNexus web page would contain the “details” of the data -- schedules, place, and time -- and a URL of the actual event.
7.0 Evaluation and Conclusions

7.1 Summary

The vast data available on the Internet allows the client a large amount of new information yet one’s ability to understand patterns and connect information to one’s environment is not augmented in tandem. A context-aware approach could be a new phase of the internet’s evolution which, like any evolutionary process, is shaped by an array of factors with opportunities to be either exploited or missed; context-aware computer applications emphasize the relationship between the individual and behavioral setting. MapNexus explores this future by elucidating a framework to define a relationship between architectural space and the range of human behavior expected within the walls of a particular building. By rethinking the purpose of “the web” and focusing intently upon the “potential energy” of architectural spaces, the ensuing possibilities for human behavior can be modelled. MapNexus thus aims at providing the client web-based information germane to a given spatial/temporal context to allow immediate interpretation with minimum ambiguity. The goal is not to utilize the web to look for obscure data but rather to utilize the web to provide data that could be used in one’s immediate context to help make every-day decisions.

In a WAN like MIT, an urban environment can be geographically tessellated into a series of polygons reflecting the radio transmission range of each AP. By using these APs as spatial aggregation units in a relational database, logical assumptions about short-term behavioral patterns can be modelled to capture, encode, and interpret context-aware cues. These cues stored in a RDBMS that through a series of SQL queries and related algorithms produces fully context-aware output that fits the client’s short-term activity landscape and preference pattern. The method does not require any special client-side software and preserves the user’s anonymity as it derives the client’s physical location implicitly by connecting behavior patterns and wi-fi tesselations. Such a framework permits a new type of context-aware web interaction based upon “common sense” patterns that are specific to MapNexus.
architectural interior spaces at a given time.

MapNexus draws together three separate methods to represent knowledge about architectural interior space and expected behavioral patterns: from GIS the “spatial aggregation unit;” from AI the “context-aware engine;” and from NLP the multi-dimensional context markup from OpenCyc, a framework to create a grammar to represent common sense knowledge propositions. Seeking and making sense of the vast amount of information on the web requires a cognitive ability to place the pieces of the puzzle together in order to complete the task. MapNexus uses the concept of a zoning map with its table of “permitted uses” to represent human behaviors in the short term as a function of interior architecture – the land use of buildings and rooms. In zoning maps, each land-use has a table of permitted uses – activities that people may or may not engage in on this particular premise. Likewise each building has different functions within an urban fabric. MapNexus uses a coherent scheme of preference representation of the MIT campus based on logical deductions about a building’s use to create context profiles.

MapNexus’s goal is not lofty, but mundane – helping one get public transport schedules or suggesting things. Yet the goals are achievable and represent the sort of “connectionist” approach to application design by applying Minsky’s theories to the wireless technologies. Once assumptions about the client’s disposition and point of view can be made, MapNexus can then filter event information passing through via a series of SQL queries and related algorithms that compare the “preference profile” for each event as a function of the client’s space and time. Once the events are ranking in order of relevance to the client’s context profile, the events are displayed on the MapNexus homepage in an ordered fashion. This method of “event matching” to the client’s context allows for comparisons to be made even when events are vastly different types.

7.2 The MapNexus Advantages

The three primary advantages of MapNexus are (1) its ability to implicitly capture context, (2) its focus on the user’s short-term intent, and (3) its straightforward database
structure. The framework is based on the ability to identify accurately the location and function of spatial aggregation units on the MIT campus, as well as behavioral patterns of the intended audience – the MIT undergraduate population. As it draws directly from the structure of the wireless WAN, it requires no cookies, login sequences, or downloads. With MapNexus the client is anonymous; it is the AP that is tracked, thus preserving client’s privacy. The system is not complex compared to far more sophisticated Natural Language Processing algorithms, which must be maintained by a single ‘context-master’ rather than multiple web masters. Allowing implicit data sets to be presented in context with our physical environment could be a significant addition to the web. Moreover, there is utility in implicitly modelling data relationships as a function of time and place because computers are more central to our life and the need to integrate processes becomes more important.

7.3 The MapNexus Framework’s Disadvantages

The principle disadvantage of the MapNexus framework is that the framework is not easily transferable to other contexts where context-aware applications are readily sought, such as a heterogeneous urban environment. Much of MapNexus’s “intelligence” derives from the fact that MIT is a closed community, and that only people who have an official MIT network account can utilize this intranet site. Thus the people being modelled are a sub-set of the larger population, and they are moving over a space rich in architectural cues. MapNexus depends upon the context-master to determine the client’s intent by modelling. Therefore any criticism of function is bound to be tied to a single person’s “view” of the campus. While the MapNexus framework is a ‘one size fits all’ approach, based on a precise knowledge of the physical environment and a homogeneous audience within the system, this very framework would not be as cognizant of a client’s intent in a less predictable environment. Furthermore the system is quite complex and would require expert maintenance; the system may well prove to complicated for all but the most advanced web designers to implement.
7.4 Conclusion

As the wireless WAN gets larger, there is a growing need for information relevant to a client's location and the specific time of request. MapNexus is a framework for improving the information provided to the client by gathering cues from the user's physical environment. Since wi-fi is increasingly popular, it would follow that a client's increasing mobility would lead to a shift in how one approaches information seeking; a return to the traditional form of communication (speaking in context) may yield clues as to how to improve the page of the future. If the web has become too big with too much information to sort through, it then becomes our task to make it smaller and more manageable again — to localize it. The next phase in this project would be to create a prototype in a part of the campus; only with deployment of the concept with a WAN followed by extensive user testing can the practicality and usefulness of the ideas be tested adequately.
REFERENCES


Cohen, Philip and Sharon Oviatt, 2000, “Multimodal Interfaces that Process What Comes Naturally,” Communications of the ACM. March/Nov. 43, No. 3.


