Published in the Proceedings of CHI '99, May 15-20, 1999.

Urp: A Luminous-Tangible Workbench for Urban Planning and Design

John Underkoffler and Hiroshi Ishii
MIT Media Laboratory, Tangible Media Group
Cambridge, MA
{jh,ishii}@media.mit.edu

ABSTRACT

We introduce a system for urban planning – called *Urp* – that integrates functions addressing a broad range of the field's concerns into a single, physically based workbench setting. The *I/O Bulb* infrastructure on which the application is based allows physical architectural models placed on an ordinary table surface to cast shadows accurate for arbitrary times of day; to throw reflections off glass facade surfaces; to affect a real-time and visually coincident simulation of pedestrian-level windflow; and so on.

We then use comparisons among *Urp* and several earlier *I/O Bulb* applications as the basis for an understanding of *luminous-tangible interactions*, which result whenever an interface distributes meaning and functionality between physical objects and visual information projectively coupled to those objects. Finally, we briefly discuss two issues common to all such systems, offering them as informal thought-tools for the design and analysis of luminous-tangible interfaces.

Keywords

urban design, urban planning, architectural simulation, luminous-tangible interface, direct manipulation, augmented reality, prototyping tool, interactive projection, tangible bits

SCENARIO

Two urban planners, charged with the design of a new plaza, unroll onto a large table a map showing the portion of the city that will contain their project. They place an architectural model of one of the site's buildings onto the map. Immediately a long shadow appears, registered precisely to the base of the model, and tracks along with it as it is moved. They bring a second building model to the table and position it on the opposite side of a large fountain from the first; it too casts an accurate shadow. "Try early morning," requests one of the planners. Her colleague places a simple clock on the map; a glowing '3pm' appears on the clock's face. The colleague rotates the hour hand around to seven o'clock, and as '3pm' changes to a luminous '7am' the shadows cast by the two models swing around from east to west.

It is now apparent that in the morning the second building is entirely shadowed by the first and will receive no direct sunlight. The urban planners decide to try moving the first building south by eighty yards, and upon doing so can immediately see that this solution restores the second building's view of the sun. The just-moved building is now only twenty yards to the north of an east-west highway that borders the plaza on the south; one of the planners places a long road-like strip of plastic on top of the map's representation of the highway, and tiny projected cars begin progressing at various speeds along its four lanes. The other planner brings a wand into contact with the nearby building, and the model's facade, now transformed to glass, throws a bright reflection onto the ground in addition to (but in the opposite direction from) its existing shadow. "We're blinding the oncoming rush-hour traffic for about ninety yards here at 7 AM," he observes. "Can we get away with a little rotation?" They rotate the building by less than five degrees and find that the effect on the sun's reflection is dramatic: it has gone from covering a long stretch of highway to running just parallel to it.

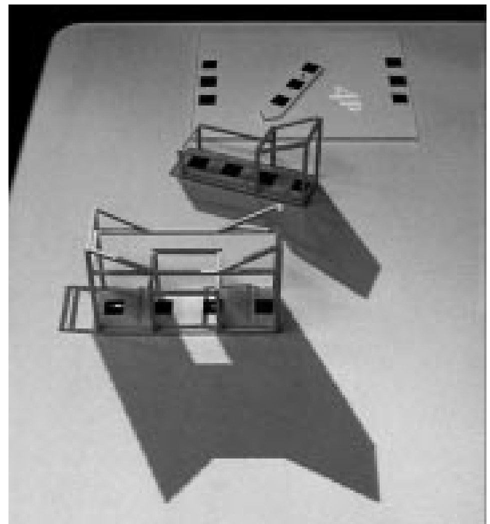

FIGURE 1: URP, SHOWING LATE-AFTERNOON SHADOWS

The urban planners position a third building, near and at an angle to the first. They deposit a wind-generating tool on the table, orienting it toward the northeast (the prevalent wind direction for the part of the city in question). Immediately a graphical representation of the wind, flowing from southwest to northeast, is overlaid on the site; the simulation that creates the visual flow takes into account the building structures present, around which airflow is now clearly being diverted. In fact, it seems that the wind velocity between the two adjacent buildings is quite high. The planners verify this with a probe-like tool, at whose tip the instantaneous speed is shown. Indeed, between the buildings the wind speed hovers at roughly twenty miles per hour. They slightly rotate the third building, and can immediately see more of the wind being diverted to its other side; the flow between the two structures subsides.

INTRODUCTION

The scenario above depicts the use of *Urp*, a working application for urban planning. Like *Illuminating Light* (its more primitive predecessor) *Urp* is built atop the *I/O Bulb* infrastructure and employs the *glimpser*-and-*voodoo* vision analysis pipeline [5] to identify and locate its component objects. Both applications also demonstrate *luminous-tangible interaction*, a style in which a participant's relations with the system consist of manipulation of physical objects and the resultant ongoing projection of visual information onto and around these same objects; indeed, *Urp* extends the variety of such interactions, as we will see later.

The paper has two principal parts: in the first, we describe *Urp*. This entails a brief introduction to the collection of concerns in the urban planning domain that motivate the present work, including a review of some traditional means of addressing these concerns; a recapitulation of basic material introduced elsewhere regarding the *I/O Bulb* and *Luminous Room* infrastructures that make the *Urp* application possible; and finally the implementation issues and a function-by-function description of the *Urp* system itself.

The second part begins with short descriptions of several other projects built with *I/O Bulb* technology (some of which have not yet been otherwise published or publicly presented) and uses a comparison among these and *Urp* to suggest two 'Luminous-Tangible Issues', early thought-tools for the design and analysis of systems that subscribe to luminous-tangible interaction styles.

BACKGROUND

The domain of urban planning involves the relationship between architectural structures and existing settings (to harshly distill what is of course a very complex field).

Urban Planning Issues

The work reported here focuses in particular on the arrangement of architectural forms with the goal of fulfilling certain aesthetic goals while at the same time respecting a variety of practical constraints. Among the primary constraints we will consider are the following:

· **shadows:** Does the proposed placement of a tall structure mean that from dawn until 10 AM no direct sunlight will reach an existing building that was formerly able to see the sunrise? Could such a situation be the source of a lawsuit? (Yes, in fact.)

· **proximities:** Is a building too close to a roadway? Is the distance between two adjacent buildings too small to allow adequate pedestrian flow? Is a building too far from an intersection?

· **reflections:** When a building with all-glass sides is erected as proposed, will low-angle sunlight (in early morning or late afternoon) be reflected directly into the eyes of oncoming motorists on a nearby highway? For what distance along the highway will this glare be present?

· **wind:** Does the placement of a building into an existing urban configuration result in a constant 80 km/h airflow over its north face? Does it result in a low-pressure zone on its east side that will make opening doors difficult?

· **visual space:** How will what pedestrians see change with the addition of the new structure? Will the space become visually claustrophobic? Will the new structure introduce a pleasing regularity into the skyline?

Standard Options

A collection of traditional techniques exists for the treatment of these different constraints. Shadow studies are often undertaken by placing a specially-mounted light source above a model of the site in question; the exact position of the source is determined by consulting a table indexed through time of day, season, and latitude. This scheme is somewhat arduous, difficult to adjust, and ultimately not quite correct (the source throws shadows from a finite distance, while the true sun's rays are essentially parallel as they reach our planet). Distances are of course easy to measure by hand. Reflections present further difficulties, however: adapting the shadow-technique (light sources positioned above the models) for reflections requires placing small patches of reflective material the models' various surfaces, but the difficulty of obtaining extreme flatness and full registration of these patches makes accurate results less than likely. Each of these concerns can also of course be addressed solely on paper using drafting techniques that involve tedious constructions and by-hand calculations [7].

Airflow analysis is another proposition altogether. Here, the only viable non-computational approach is to immerse the model or models in a wind tunnel; smoke sources released upstream from the subjects can be used to show overall flow patterns. No matter the level of detail imposed on this kind of setup, however, the actual scale of the phenomenon being tested differs from that of the simulated setting – fluid dynamics is sensitive to scale – so that observations are valid only to a certain extent.

More recently, computational approaches to each of these analyses have become available. There are several CAD-style architectural applications (AllPlan FT, ArchiCAD, 3D Studio Max, AccuRender, etc.) that incorporate on-screen facilities for shadow and reflection studies. Airflow simulation is still a difficult matter; full solutions to the prevailing *Navier-Stokes* equations are always expensive, and no known system allows real-time rearrangement of architectural structures within the ongoing simulated flow field.

IMPLEMENTATION

It was our intent to construct an interactive workbench for urban design and planning that would collect together functions addressing the concerns listed above; the novel aspect of our system would be that its information would all be centered on or attached to actual physical models of the architecture in question. The result of this effort is *Urp*.

I/O Bulb & Luminous Room

The large-scale goal behind the work that has led to *Urp* and its companion systems is the wholesale transformation

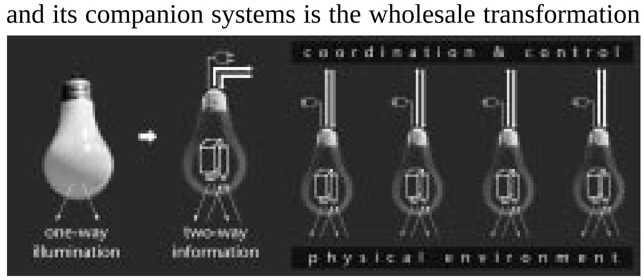

FIGURE 2: I/O BULB (L) AND LUMINOUS ROOM (R) CONCEPTS

of architectural space – to make of each surface an information-display-and-interaction structure. The approach we have been pursuing calls for the conceptual generalization of the familiar lightbulb into the *I/O Bulb*, as follows: if an ordinary incandescent bulb is actually a low-resolution digital projector – specifically, 1x1 pixel(s) – then we increase this resolution, so that the lightbulb is capable of projecting images into the space around it. At the same time we incorporate a tiny video camera that looks out at the world around the bulb. The resulting structure, called an *I/O Bulb*, is capable of simultaneous optical input and output. The work described here makes use of a prototype *I/O Bulb* constructed with commercially available projectors and cameras.

The notion of a *Luminous Room* extrapolates from just one to a collection of many *I/O Bulb*s, computationally interlinked and distributed throughout an interior architectural space. The resulting aggregate of two-way optical nodes addresses every portion of a room, and is thus one way of achieving our original space-transformation goal [6].

Software Components

glimpser & voodoo

Currently, *I/O Bulb* applications like *Urp* that need to identify and locate specific, known objects use an optical tagging scheme in which small colored dots are applied to the surface of each physical implement. A simple, low-level machine vision system called *glimpser* is used to find all colored dots of some specified size within the video input stream supplied to it by the *I/O Bulb*. For each video frame, *glimpser* passes a list of whatever dots it has found to *voodoo*, with which it communicates over the network as a client-server pair. *voodoo* is a software tool whose job it is to recognize among each amorphous collection of dots as many known patterns as possible; these patterns have been defined by the end application that *voodoo* serves (here, *Urp*). Affixing the appropriate pattern of actual colored dots to each object is then all that is required for applications to track it using the *glimpser-voodoo* pipeline [5].

wind simulation

We employ a variety of cellular automaton called a 'lattice gas' [2] to simulate pedestrian-level airflow through *Urp*'s workspace. The lattice gas computation involves a grid of hexagonal cells, each of which can support up to six gas 'particles' – one for each face. The state of each hex-cell is represented at every instant as a modest six bits: if a bit is on it implies the presence of an incoming particle, understood as travelling toward the center of the cell through that bit's corresponding side. At each timestep, every cell is 'reacted' according to a small set of rules that determine whether and how particle collisions occur within a cell; the rules are arranged to preserve momentum. After reaction, the redirected particles from each cell undergo transport to the boundaries of the six surrounding cells, and the cycle then repeats.

We use a 100x100 grid of lattice gas cells to simulate windflow in the workspace. The motions from contiguous 4x4 sub-blocks of cells are averaged to find an aggregate flow: local wind direction and magnitude. Obstacles – i.e. the bases of buildings – are represented by 'filling in' the appropriate cells, disallowing them from containing particles and causing incident particles to bounce directly back from their boundaries. Meanwhile, because such a small grid displays preferential anisotropy along its three major axes, it's not possible to represent arbitrary flow directions accurately. Instead, the grid is held fixed, with particles injected from the right side flowing leftward, while the world (i.e. building footprints) is rotated opposite the intended wind direction and analyzed into the grid. The resulting simulation is then rotated back once more (so that the airflow is moving in the originally specified direction) and projected down into alignment with *Urp*'s objects.

Functions & Requirements

Shadows

The shadow-casting facility was the first portion of *Urp* to be constructed, and was in fact the original catalyst for thinking about the field of urban planning: we'd asked ourselves "what if little models of buildings could cast adjustable solar shadows?". This function is very simple; any building placed in the working environment continuously casts a shadow, and the sole influence available to the urban planner is a clock, whose instantaneous setting determines the time of day and thus the position of the computational sun (see Fig. 1). If the clock object is removed from the workspace, time is 'locked' at its most recent value.

An early incarnation of the shadow function allowed time to jump instantaneously between different values as the clock – quantized at every-hour-on-the-hour values – was adjusted. The resulting visual discontinuity was somewhat disconcerting, particularly during rapid changes from mid-morning to mid-afternoon: the shadow appeared to flop around in a way that (wrongly) suggested inaccuracy. Particularly when compounded with the inevitable small positional uncertainties that result from (genuine) video-noise-based imprecisions in our machine vision pipeline, this proved fairly confusing. Instead, the current system interpolates from one time value to the next using a cubic spline (the transition lasts about one second). This gives rise to appealing shadow transitions, whose smooth 'swinging' motions strongly recall time-lapse cinematography.

Distance Measurements

An initial test in which every building and road structure constantly displayed its distance from every other left the workspace far too cluttered and visually distracting. Rather,

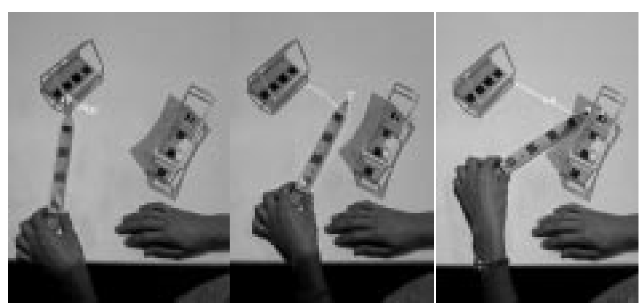

FIGURE 3: TAKING A DISTANCE MEASUREMENT

Urp now provides a distance-tool (shaped like a pencil but with the image of a ruler stretching between the pencil tip and eraser) that can be used to connect together selected structures. To do this, an urban planner touches the tool's tip to one building, on which one end of a sinuous line is then anchored; pulling the tip-end of the line away and eventually touching a second building or a road then connects the two structures, the line's curves flattening to leave it straight. A display of the line's length floats along and around it, and this number continuously changes as the connected structures are moved. When this display is no longer desired, touching the eraser end of the tool to either connection point disconnects the line.

Reflections

Long, thin *voodoo*-tagged strips represent roads; placing these in the environment engages a traffic simulation, whose automotive components are projected onto the plastic strips. Crossing two strips at any angle automatically generates an intersection with implicit traffic-control signals, so that cars come to a standstill in one direction while cross-traffic flows.

A transparent wand placed onto the table shows a **B** at one end and a **G** at the other. Touching the **G** end of the wand to any building causes its facades to become glass, so that

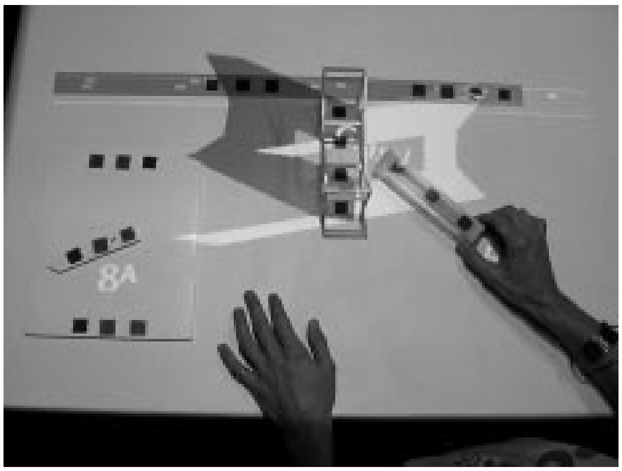

FIGURE 4: A BUILDING BECOMES GLASS

solar reflections are generated and projected onto the ground. It is apparent that reflections are far less intuitive for most people than are shadows – in part because of the angle-doubling that occurs at the bounce surface, and in part because not all of the components of the reflection are necessarily in contact with the object itself: some small ‘polygons of light’ can be thrown huge distances away from the building that generates them, depending on the angle and orientation of the responsible surface.

Incidence of reflected sunlight onto the various roadways is always immediately evident, and it is easy to experiment with the small angular adjustments that give rise to large changes in these reflected patterns. Finally, touching the **B** end of the wand to a glass building transforms its facades back into brick, and the reflections disappear.

Wind Effects

Urp’s airflow simulation is engaged simply by placing the wind-tool – a kind of inverse weather vane – anywhere on the table; orienting the tool selects one of eight quantized directions (the eight major compass points). The simulation is displayed as a regular array of white segments, whose direction and length correspond to the instantaneous direction and magnitude of the wind at that position. In addition, ten red contour lines are shown, generated simply by ‘connecting the dots’ from location to location according to the local field vectors. These displays take a qualitative form; for more precise measurements, the anemometer-object is available. Placing this arrow-shaped tool within the field samples and numerically displays the flow magnitude at the precise position of the tool’s tip. Periodically, these numbers break off from the tool and go floating through the field as a further means of depicting larger-scale flow patterns.

Although the airflow simulation is the most computationally expensive part of *Urp*, the entire system remains usably interactive and responsive at a modest eight Hertz – so it’s possible to move buildings around the workspace and immediately view the effects on wind flow.

Site Views

The most recently added functionality provides a mechanism for ‘previewing’ a configuration of buildings from various points of view. Since the model buildings’ three-dimensional forms are already resident in the system (necessary for the calculation of shadows), it is a simple matter to render them in perspective and with simple shading parameters. A camera object is provided for this purpose; driving this camera about the workspace results in the updating of a real-time rendering of the current arrangement of buildings in the site, as viewed from pedestrian height and the position and orientation of the camera.

Objects

Irrespective of the range of functions attached to them (investigation of which is the topic of the latter half of this paper), the *forms* of the various physical elements employed in *Urp* rove through a small part of an object-design space. The architectural models, of course, have well-dictated forms: the system is predicated on the idea of attaching variegated graphical information to *pre-existing* models. The road-object, too, must correspond at least in its dimensions to the simulation that will be overlaid on it.

For the remainder of the objects, however, no particular form is necessarily prescribed. Some, like the wind-tool and the distance-measuring-object, attempt to denote their

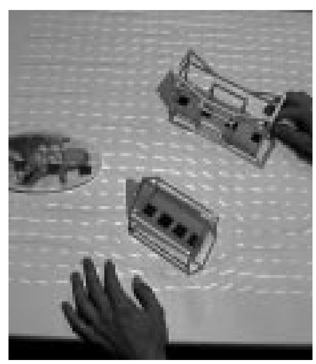

FIGURE 5: WIND

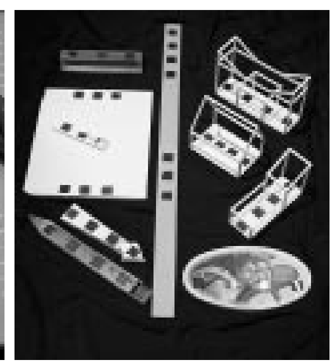

FIGURE 6: STRUCTURES & TOOLS

function through suggestive pictorial elements. Others, including the clock-, anemometer-, and material-transformation-objects, are abstract in form and hint only vaguely at their intended use. In short, no specific design methodology has yet emerged or been chosen.

But as we build more and more *I/O Bulb* applications, and as the accessible complexity of each increases, objects will unavoidably multiply. Without yet addressing the problems of this inevitable overpopulation, we acknowledge that the general issue of how object form is related to object meaning is an important one. It may be that a measure of standardization is called for, so that a recognized vocabulary of object appearances imposes some order on design; alternately, the application designer may be free to assemble arbitrary forms, with the understanding that end users of any system are necessarily semi-expert and thus expected to have learned its individual ‘language’.

EXPERIENCE & DISCUSSION

Informal Experience

While we have not yet subjected *Urp* to formal user testing (which is planned as part of a future collaboration with architecture students), it is worth noting in the meantime the reactions of the many people who’ve already been able to experiment with the system: general attitudes toward *Urp*’s new interface style and specific comments about its functionality are already helping us to understand and refine this and other such systems.

Professionals

Close to two dozen architects and urban planners (both practicing and academic) have either watched demonstrations of or directly experimented with *Urp*. Their overall impressions have been uniformly favorable; critically, most of the professional visitors said that they would voluntarily begin using a system like *Urp* immediately if it were available. Academicians affirmed its usefulness for teaching and ‘quick turnaround’ student prototyping. Practicing architects mentioned that not only would the system aid in their own personal design efforts, but that it could be invaluable for client presentations (in which, at the moment, the activity of viewing physical models and the activity of viewing animations and simulations of light & shadow, windflow, etc. are always separate). Further, several younger professionals stated that such an application would help them to communicate ideas to seasoned, older practitioners within their firm (especially founders!) who have otherwise resisted attempts to ‘computerize’ aspects of their art.

Many commented that it was unusual and significant to find so many of the field’s major concerns addressed by a single application, and all responded excitedly to the use of the architectural models themselves as the system’s principal ‘interface’. One insider was particularly delighted at seeing wireframe architectural models cast solid shadows, while insisting “and yet it doesn’t bother me at all – the shadows are entirely believable”.

Others

Perhaps as many as two hundred visitors with no special expertise in the urban planning field have also observed or directly operated *Urp*. The easy and universal familiarity of architecture apparently minimizes the ‘domain knowledge hurdle’, allowing these nonprofessional experimenters to be strongly (and fearlessly) engaged by the system. Several asked about an expanded functionality that could encompass not just the phenomena of interest to urban planners but also other distinctly nonphysical processes to be simulated and attached to the geometric distribution of structures in *Urp*. Questions arose about economic simulations (what’s the effect if the bank or the post office is twice as far away, or is turned wrong-way-round so that the door is on the other side?) and production-flow simulations (can we increase efficiency by building a second warehouse and interposing it between the manufacturing plant and the shipping building?).

Others took a larger conceptual leap, generalizing *Urp*’s capacities to suggest similar treatment of their own domains’ problems: “What about a luminous-tangible tool for design of office spaces?”, “Could we build a system to interactively simulate ventilation flow patterns throughout a theater?”, and so on.

Known Problems

A small shortcoming of our object-mediated interaction style becomes apparent through the use of *Urp*’s site-view camera. Because an object with physical extent (i.e. the camera object) must be employed to designate the desired position and orientation of the view to be rendered, it’s simply not possible to get immediately next to an existing structure. That is, if we want to see a rendering of an architectural structure in some proposed location as viewed from, say, the doorway of another building, we’d need to place the camera object closer to the building object than the physical extents of both together will allow. In the real world, of course, this is no problem at all because of the vastly different scales of a building and a camera. Inside our simulation world, however, all objects and tools must be represented at essentially the same scale.

So the same properties of physical objects that are advantageous in some circumstances (e.g. three-dimensional collision detection is computationally expensive, but the impossibility of interpenetrating *Urp*’s architectural models is a convenient constraint that automatically mirrors the desired impossibility in the real situation) can simultaneously be detrimental in other circumstances (our inability to position the *Urp*-camera ‘in the doorway’ of a building, when that would present no difficulty for a real camera).

The lattice gas used to simulate airflow in *Urp* – while a true Navier-Stokes solution – is admittedly inappropriate in several ways. Most important is that we use a two-dimensional structure to approximate what is ultimately a three-dimensional phenomenon: *Urp* ‘air’ flows sideways, but can never flow *up*. The scale of the simulation is incorrect as well; with the grid dimensions we are constrained to (in the interests of real-time operation), what is simulated is closer to the micron domain than the meter domain. This scale mismatch then has implications for resulting fluid properties, including viscosity and Reynolds number.

FUTURE

Efforts are already under way to construct two additional *Urp* workspaces for a new design studio in MIT’s architecture school, where they are to be used as a teaching tool and for student experiments. We intend to take this opportunity to simultaneously pursue formal user-testing studies.

Based also on comments from professional architects and urban planners, we are considering an expansion of each of *Urp*’s individual functions, by way of bringing the application nearer to ‘actual usability’. Many such enhancements are immediately evident: built-in zoning knowledge, so that automatic warnings are generated when proximity or other positional violations occur; additional controls for specifying latitude and season; a light-and-shadow integration tool that will cause the cumulative light incident over a year’s time to be calculated and displayed within the workspace, as an aid to landscape architects; and the incorporation of topographic information, so that non-planar sites can be accurately treated.

It will also be important to introduce a facility for projecting ‘absent’ components into the workspace: buildings that are part of the site but for which no model is available, or whose positions cannot be changed by the planner. These elements would of course still cast shadows and exhibit the various forms of interaction enjoyed by the physically present models.

Such projection-only components may also represent real models manipulated by colleagues at a remote location with whom the urban planner is collaborating. A distributed version of *voodoo* (an important software modification for the *Luminous Room* infrastructure) will allow planners at distributed *Urp* installations to collaborate directly: objects manipulated at each location will be projectively represented at the other. These remote collaboration functions will be incorporated into and tested in the new *Urp* workspaces being constructed for MIT's architecture studios.

OTHER LUMINOUS-TANGIBLE SYSTEMS

We have begun to analyze our observations and experiences in constructing luminous-tangible applications; the issues that seem invariant across these different systems – *Luminous-Tangible Issues*, perhaps – are slowly emerging. We review here several other *I/O-Bulb*-based projects, followed then by a brief introduction to two of these issues.

Context

Illuminating Light

An earlier application constructed with the *I/O Bulb* is *Illuminating Light*, which allows engineers and students to prototype laser-based optical layouts. The system provides an assortment of models representing simple optical elements, including lasers, mirrors, lenses, beamsplitters, recording film, and so on. Each of these objects carefully recapitulates the function of the element of which it's a model, so that a laser placed on the table under the *I/O Bulb* appears to emit a precisely aligned beam; a beamsplitter placed in this beam transmits half and reflects half; and a lens breaks an incident beam into a diverging fan of sub-beams.

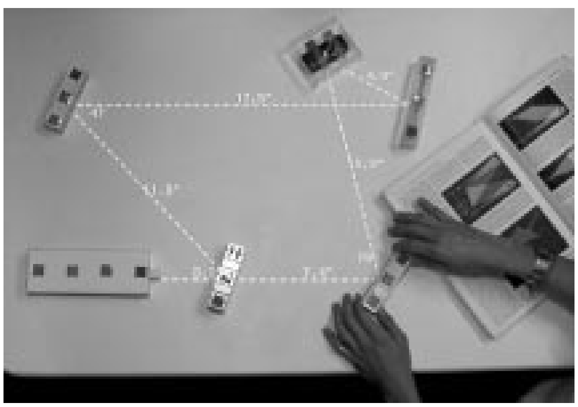

FIGURE 7: ILLUMINATING LIGHT

Illuminating Light depends, like *Urp*, on the *voodoo*-tagging of its objects with colored dots and on the cooperation of a *glimpser* / *voodoo* machine vision pipeline. Again, the only 'tools' available in the system are faux optics; and although the display of ancillary qualitative information is automatically projected into the real-world setup, no objects are provided for explicit measurement or 'higher-level' modification of the layout being constructed. In this way the application closely mimics a corresponding real-world optical engineering environment, in which the only access to control of light propagation is through the manipulation of physical optics.

Seep

The first *I/O Bulb* application to be built without the use of *glimpser* and *voodoo* is a simple fluid dynamics workbench called *seep*. The same lattice-gas simulation deployed in *Urp* runs here, but instead of taking as input the position and orientation of structures known in advance (i.e. *Urp*'s various architectural forms), *seep* allows arbitrary objects to be placed in the flow path. The shapes of these objects are extracted from the visual field captured by the *I/O Bulb* using rudimentary frame-differencing techniques; these silhouette shapes then serve as obstacles appropriately positioned within the flow simulation's boundary.

FIGURE 8: SEEP: FLUID FLOW WITH ARBITRARY OBJECTS

The result is a real-time simulation in which fluid appears to flow from right to left across a table surface; any object (non-inanimates like hands are also valid) placed on the table rapidly diverts the flow, which for example exhibits increased field velocities in certain places – as one would expect – in order to maintain the overall right-to-left flux. Moving the obstacle-objects produces beautiful and fluid-dynamically satisfying transient effects, including slipstreams, eddies, sloshing, and all manner of swirls. Although *seep* is in no sense a complete application – there's no facility for extracting quantitative measurements, or for affecting the simulated flow constants, for example – it is a promising new kind of tool for providing intuition for complex physical phenomena and their interaction with real-world objects.

Early Chess & Bottle System

The earliest luminous-tangible application – built with an *I/O Bulb* aimed horizontally to treat an entire wall in a small office – collected together a few 'toy' functions. The

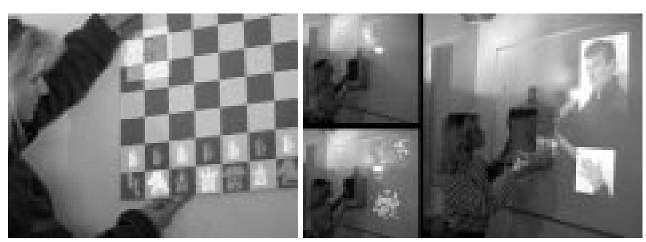

FIGURE 9: WALL CHESS & BOTTLE STORAGE

system recognized a large chessboard that, when brought into the space, would be gradually populated by animated projective chesspieces; the thus-far-unrealized further intent was that physical pieces placed on the board could be identified and located, allowing a half-physical, half-luminous game to be played. Meanwhile (and simultaneously, if desired), a large bottle was able to act as a container for digital information: text, images, and live video could be placed inside the bottle which, irrespective of subsequent movement about the space, could always be made to disgorge these contents. Finally, a colored paddle was available for most of the actual manipulations in the system; it was with this paddle that sample documents could be created, moved, disposed of (in a physical trash can), placed

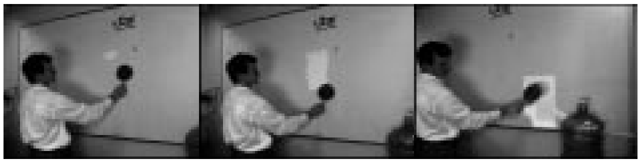

FIGURE 10: PADDLE CREATES & MANIPULATES DOCUMENT

into or pulled out of the bottle, and so on.

DISCUSSION

Object Meanings

What are the different ways in which a luminous-tangible system can understand or make use of objects? We offer an

FIGURE 11: CONTINUUM OF OBJECT MEANINGS

analysis space that arrays all possible interpretations along an axis that moves away, in both directions, from a center representing a maximally 'real-world' object reading.

Note that these classifications are intended to apply only to objects considered in the context of a luminous-tangible system – we are not attempting a generic scheme appropriate for arbitrary TUIs (tangible user interfaces) [3]. Moreover, we are not proposing a formal grammar (as does Ullmer in [4]) for the analysis of TUI-based object-to-object interactions; the Object Meanings axis classifies *individual* objects. Finally, it must be understood that we use the words 'noun' and 'verb' merely as a convenient way to suggest certain properties, and not in any attempt to imply a full mapping between luminous-tangible objects and linguistic parts of speech (as is undertaken in [1]).

Object As Noun

These objects occupy the center of the axis and are likely the most obvious in their behavior. They are fully literal, in the sense that they work in their luminous-tangible context very much the way objects 'operate' in the real world – an Object As Noun exists in our applications simply as a representation of itself: an immutable thing, a stand-in for some extant or imaginable part of the real-world. All the objects in the *Illuminating Light* application are of this type – each of the optics models is meant to be understood (in function) as its real-world counterpart. The buildings and roads in *Urp* are also of this variety.

Object As Verb

As we move to the right along the continuum, away from Object As Noun, inherent object meaning is progressively abstracted in favor of further – and more general – functionality. The material-changing wand in *Urp*, for example, is an Object As Verb. It is not understood as 'present' in the world of *Urp*'s simulation, but exists to act on other components that are, or on the environment as a whole. The clock and wind objects do just this, in affecting ambient conditions like time, solar angle, and wind direction. However, both these tools in fact lie somewhere along the continuum between Object As Noun and Object As Verb, inasmuch as they are each, in part, a metonymic proxy for objects that *do* conceptually occupy the simulation's world – i.e., the sun and the aggregate phenomenon of 'wind'.

Object As Reconfigurable Tool

This variety of object-function is fully abstracted away from 'objecthood', in a way perhaps loosely analogous to a GUI's mouse-plus-pointer. The paddle in the chess-and-bottle system is of this sort, but where a WIMP-style interface typically uses a series of menus to change the function of the mouse, the paddle depends for these meaning-alterations on context and state. Since that single early use of this kind of object, however, we have temporarily avoided its further deployment: to simply transplant some variation on the mouse-and-menu idea into our applications is too easy, and would fly in the face of the basic tenets of building luminous-tangible systems in the first place. We do believe that there exists a proper (non-menu) method for introducing such reconfigurable objects into the world of the *I/O Bulb* – and this solution will soon be required to combat the inevitable proliferation of objects that results from constructing ever more complex applications.

Object As Attribute

As we move to the left away from the center of the axis, an object is stripped of all but one of its properties, and it is this single remaining attribute that is alone considered by the system. The arbitrary objects that act as flow obstacles in the *seep* application are one example: there, nothing matters but the *shape* of what's placed in the workspace; all other attributes of the objects used are ignored. Another system might consider (for some purpose or other) only the color of an object, or the object's size, or its velocity.

Object As Pure Object

This last category is the most extreme, and represents the final step in the process of stripping an object of more and more of its intrinsic meanings. In this case, all that matters to a luminous-tangible system is that the object is knowable as *an object* (as distinct from *nothing*). It may or may not be important that the object be uniquely identifiable; to take an example in which it is, we can imagine extending the digital-storage-in-physical-bottle scenario to a full Luminous Room setting in which information can be stored in arbitrary objects, wherever we may happen to be. Thus, just as we might scribble a phone number on anything nearby – an envelope, a magazine, even a hand – the following scenario would make sense: "Where did you put the directions to the restaurant?" "Oh – they're in the scissors."

The scissors don't matter as scissors; all that's relevant is that they exist and are distinct from other objects that might have been used instead – and that they're where the restaurant directions are.

It is at this far end of the meaning spectrum that we suddenly find that the axis is not linear, but in fact connects to itself, end-to-end: if an object has been shorn of all inherent meaning, then paradoxically it is free to be assigned an arbitrary functionality. So if we move beyond Object As

Pure Object we can find ourselves suddenly back at Object As Reconfigurable Tool.

Straddle-Balance

By definition, every luminous-tangible system locates meaning and functionality simultaneously in two contrasting places: in physical objects, which are directly manipulable by human clients of the application, and in projected digital elements, which are not. It has become apparent that the way in which an application distributes its tasks between corporeal objects and noncorporeal projection – straddling the graspable/corporeal and the digital/projective – has a great deal of bearing on its ultimate behavior and form.

The *Illuminating Light* system, for example, posed little question as to which parts of the application would be projected and which would be physical; in setting out to directly parallel the way in which optics experiments are constructed and carried out in the real world, we automatically obtained an elegant balance: physical models would represent physical optics, and projected *I/O Bulb* light would represent actual laser light. So as the real-world engineering pursuit became a luminous-tangible simulation, noncorporeal remained noncorporeal and manipulable remained manipulable. In a sense, the system very conveniently dictated its own design.

Urp represented a somewhat more complex design circumstance. However, the same pattern of solid-to-solid and nonmaterial-to-projective mappings emerged: light and shadow effects became aligned projective complements to the architectural models, as did the airflow simulation.

It is important to note that the buildings in *Urp*, through their geometric arrangement, carry no less meaning than the more 'exciting' shadows and reflections attached to them – the placement and orientation of structures is, after all, the end goal of urban planning. That is to say: in *Urp* the disposition of physical building models itself contains information; they are not just 'input' but 'output' as well.

A very different kind of meaning distribution is demonstrated by the early 'chess & bottle' system. Here, the scenario's objects carried little specialized meaning: the chessboard was simply an inert stage for the antics of the animated chesspieces, and the bottle – being a container – was essentially unrelated to the digital constructs that it contained. Instead, nearly all the functionality in the system had been concentrated into one physical tool: the color paddle. This single significant instrument was used to create documents, to move them about the space, to associate them with the bottle, to trigger the bottle to absorb them, and so on. To a certain extent, the paddle acted much like the featureless but infinitely assignable mouse of a GUI.

Clearly, applications that have very few projective components and rely mostly on physical objects lean toward 'just being the real world'; while applications that tend to ignore physical objects in favor of complex or standalone graphical components (e.g. the paddle system) encroach on familiar GUI territory. But each extreme can also be appropriate, depending on the needs it addresses and the context in which it's deployed.

Ultimately, we do not yet have a large enough body of telling luminous-tangible applications to formulate general prescriptive rules, but we can state that such straddle-balance issues will remain central to proper luminous tangible design.

CONCLUSION

We have presented *Urp*, an application for working with architectural elements in the context of urban planning and design. This luminous-tangible system attempts to address the primary concerns of this field in a novel way: by using *I/O Bulb* techniques to attach projected forms to physical architectural models, we can provide the urban planner with access to the full efficacy of computational resources in a manner that is comfortable, intuitive, and – ultimately – most appropriate given the spatial and geometric nature of the pursuit.

We have also provided a preliminary examination of luminous-tangible interactions as a general class, identifying two early issues fundamental to every such arrangement. We expect that, as more *I/O Bulb*-based applications add to the set of available examples, the current luminous-tangible issues (joined by others) will mature into a full set of proper *luminous-tangible principles*: appropriate theoretical tools for further design and analysis.

Finally, and as an aside, we are discovering that luminous-tangible interactions, apparently by their very nature, strongly engage nearly everyone. People who've played with one or more of the applications described here evince a delight in that very playing, irrespective of the task at hand. While sheer novelty surely contributes to these reactions, we also believe (for the moment leaving the assertion informal) that the proposition of giving additional meaning and animate life to ordinary inert objects is a cognitively powerful and intriguing one. So: at least as much as do benedictions from professionals in the various applications' fields, visitors' more visceral responses have begun to build a strong case for *I/O-Bulb*-mediated workbench environments, whether physics simulation (*Illuminating Light* & *seep*), design tool (*Urp* & an as yet unreported filmmaking previsualization tool), or children's construction kit (another to-be-described system).

ACKNOWLEDGEMENTS

We extend heartfelt flying buttresses to Dean William J. Mitchell of MIT's School of Architecture and Planning (high level insight & analysis); Peter Underkoffler & Andy Dahley (model design & construction); Brygg Ullmer (thinking beyond); Dan Chak & Gustavo Santos (future pillars); & Wendy Plesniak (all this & more).

REFERENCES

[1] Fishkin, K., Moran, T., and Harrison, B. Embodied User Interfaces: Towards Invisible User Interfaces, in *Proceedings of EHCI '98*, September 1998.

[2] Frisch, E., Hasslacher, B., and Pomeau, Y. Lattice-Gas Automata for the Navier-Stokes Equation. *Physical Review Letters*, 56, 1505-8

[3] Ishii, H. and Ullmer, B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, in *Proceedings of CHI '97*: 234-241, March 1997.

[4] Ullmer, B. and Ishii, H. Formal Representations for TUI Primitives: Towards a Theory of Tangible Interfaces, submitted to *CHI '99*

[5] Underkoffler, J. and Ishii, H. Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface, in *Proceedings of CHI '98*: 542-549, April 1998.

[6] Underkoffler, J. A View From The Luminous Room. *Personal Technologies*, Vol. 1, No. 2, June 1997.

[7] Yee, R. Architectural Drawing. John Wiley & Sons, 1997