Spatial Tools for Managing Personal Information Collections

Dan Bauer, Pierre Fastrez, and Jim Hollan
Distributed Cognition and Human Computer Interaction Lab
Department of Cognitive Science, University of California, San Diego
Web: *http://hci.ucsd.edu* Email: {*dsbauer, fastrez, hollan*}@*hci.ucsd.edu*

Abstract

Dynapad is a development environment designed to support research prototyping of multiscale workspaces. In this paper we describe applications designed to facilitate visual access to and spatial organization of digital photo collections and personal libraries of PDF documents. The research objective is to explore a generalization of the notion of a "pile" as a foundation for a versatile suite of tools to provide unobtrusive assistance for organizational and other sensemaking activities. We detail the architecture underlying the applications, explain how it supports diverse functionality and interaction styles, and abstract a set of principles for designing spatial tools.

1. Managing Digital Collections

People spend substantial time maintaining personal collections of varied types of digital information: photos, video clips, web bookmarks, email archives, professional documents, and other files. Our ongoing research is an effort both to understand the cognitive strategies people use in managing such collections in a visual workspace, and to build a versatile infrastructure of tools to support those strategies. Although diverse content types are often supported by different applications, our premise is that the same basic cognitive strategies likely underlie the activities of exploring and organizing any collection. This may be why spatial arrangements of elements in *piles* have proven to be such a natural and effective mechanism for managing physical desktops [16, 24, 25].

Considerable research has explored how people make use of space to organize information [14, 18]. Likewise, the utility of image-based [22, 23, 11, 13, 3] and time-based [8, 20] workspaces has drawn increasing interest. Our work continues these themes. When using spatial workspaces, people typically allocate regions of space to play specialized roles within broader activities. For example, piles can be used to categorize items or to function as reservoirs of items yet to be examined. We argue that the affordances of physical piles can be dissociated and selectively engineered in digital environments [2]. For example, while in physical piles items typically occlude each other, digital piles can be "open", with contents spread out, and thereby *remind* users of information and organizational opportunities.

Using Dynapad, the latest instance in the Pad++ lineage of zoomable interface development environments [4, 5], we are exploring spatial tools for managing many types of digital information. In this paper, we focus on digital photos and iconic representations of PDF documents such as journal articles. We demonstrate how a generalization of the "pile" metaphor [17] can serve as a foundation for new types of tools to enrich regions of a collection-management workspace with local task-specific behaviors. Throughout the paper we draw on "exploratory observations" [16, 2] of people using developing versions of our tools to organize their own collections of digital photos and research documents.

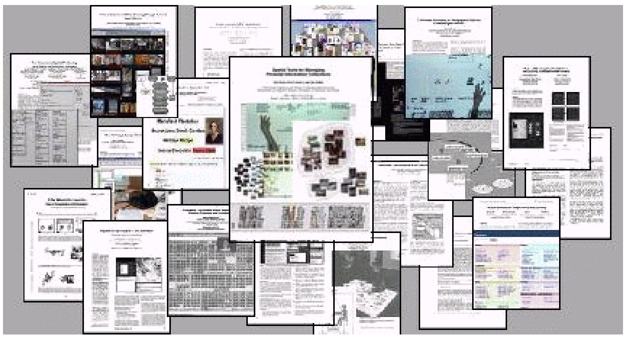

Figure 1: Sample PDF portraits of this paper and some references

1.1. Visualizing Collection Elements

Our research emphasizes the value of *visual* access to information. One content type, photographs, are already visual and are represented in a Dynapad workspace as thumbnails. Other types of content, however, are more challenging to convert to a graphical form. We have adapted Dynapad to support graphical representations of PDF documents, typically focusing on collections of research papers downloaded from the web [1]. Because images from these papers

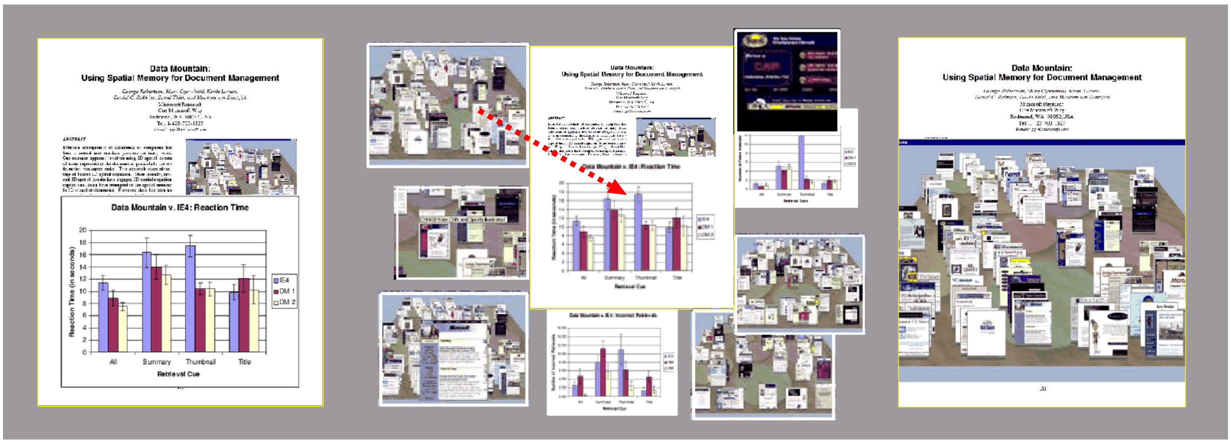

Figure 2: Editing a PDF Portrait-Collage of [22]. *Left:* An initial locked portrait. *Center:* Portrait is unlocked for editing, allowing positioning of hidden images in a new arrangement. *Right:* New arrangement is cropped and locked.

can be effective retrieval cues, we extract and collage them into "portraits" or "enriched thumbnails" [26] of the documents. Figure 1 shows sample portraits of this paper and many of its references. In Dynapad thumbnails are automatically replaced by high-resolution versions when users zoom into them. In addition, other applications can be accessed via the images (e.g., image editing programs for digital photos and PDF viewers for files associated with the portraits).

Currently, the algorithm we use to generate document portraits is relatively simple: we automatically extract all component images, sort them by file size (which reflects both image size and complexity, and therefore salience), and arrange the top few over a background image of the document's cover page. We are exploring more sophisticated strategies as well. But of course, no algorithm can always guess correctly what will be the most effective portrait for a given paper; Dynapad uses an evolving set of heuristics to make a best guess, but lets the user edit any portrait-collage, as is depicted in Figure 2. When a portrait is unlocked for editing, all component images may be moved and resized (although they are forced to stay contiguous with the background image). When locked, the collage is cropped to the boundaries of the background and unused images are "stored" out of sight and may be accessed later if the collage is again unlocked.

We are experimenting with including customizable text fragments in the portraits – for example, paper title words, keywords, or author names that can be automatically extract from PDFs. In addition, we are exploring *dynamic* portraits, whose appearance changes in different contexts (e.g., showing a slideshow of the contained images as the mouse hovers over a portrait) or at different viewing scales (e.g., using "semantic zooming" [4] to show only the largest image when zoomed out or a representative sampling when zoomed in). Finally, in new work to integrate Dynapad with the Stanford Diver system [19] for annotating audio-video source material, we have prototyped portrait-collages for video clips in which heuristically chosen keyframes serve roles similar to images from a PDF file.

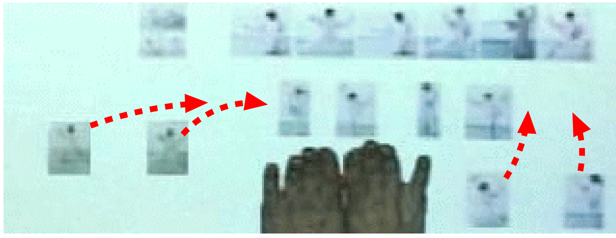

Figure 4: Manually enriching microstructure by adjoining and aligning in rows.

As we will describe below, collections of these document portraits and digital photos can be organized using a common set of arrangement tools. The future we envision is one in which regions of a multiscale workspace support effective organizations of and access to all relevant digital information associated with a project. Ethnographic researchers in our lab, for example, would like to be able to flexibly group together collections of digital photos and videos (acquired during field observations) with articles, notes, and other digital information relevant to specific research projects.

Our current system supports informal pile-based organization of collections of digital photos and portrait-collages of PDF papers. While we describe the details of operation later, Figure 3 shows a timeline lens placed over a collection of PDF documents, resulting in a temporary chronological view. Before we describe lenses and other organization-supporting tools, we briefly consider some typical examples of how people use space.

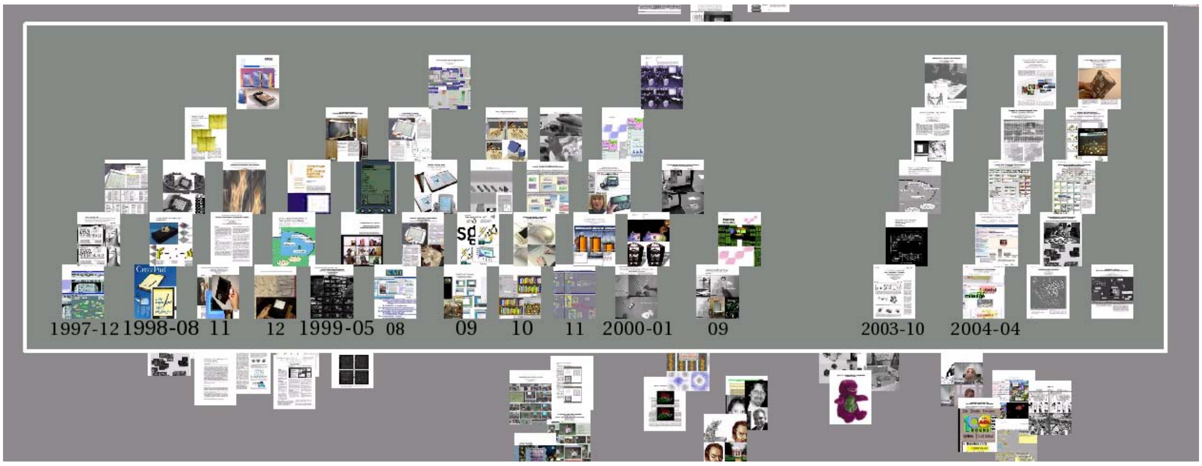

Figure 3: A *timeline lens* is positioned over document piles in a Dynapad workspace. The lens provides a temporary chronological ordering (by either PDF-creation date or date of entry into the file system) of the documents. Year and month are indicated along the bottom of the lens.

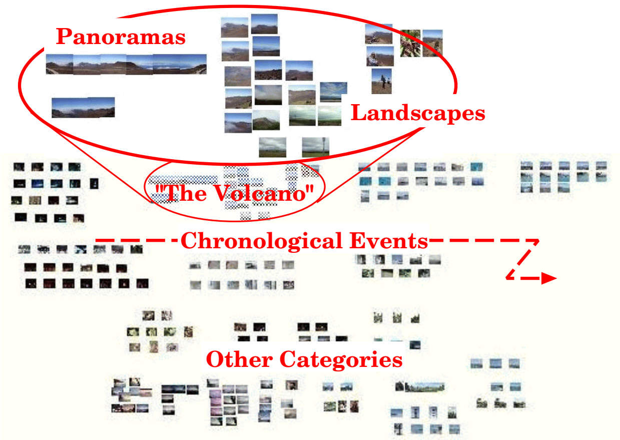

Figure 5: A richly structured workspace with sorted piles.

1.2. The Use of Space

The design of Dynapad's spatial tools is based on observations of how people use space to organize digital photos in various settings. That work is reported elsewhere [2]. Here we summarize relevant organizational strategies. Figure 4 shows one of our observed participants arranging selected photos into rows. Here she is using hand gestures to indicate "adjacent, like this". This example includes several typical aspects of participants' arrangement strategies. We will refer later to four aspects in particular:

- *Sorting.* The order of items, often in several levels at once, can convey information about one or more "parameters" of the items.
- *Separating.* Groups and categories are distinguished and emphasized by deliberately separating them with empty space, often carefully calibrated to represent information.
- *Adjoining.* Within groups, items are often brought close together (abutted, overlapping, or even stacked), both to conserve space and to facilitate iterating one's attention over them.
- *Aligning.* Items are also aligned in rows or columns, making it easy to search items serially or compare visually similar items.

In another example, Figure 5 shows another participant's workspace, invested with multiple levels of spatial structure. We emphasize its extensive use of *piles* and the four arrangement strategies listed above, both within and between piles. The upper half of the workspace has event-specific piles arranged in chronological order (left to right, top to bottom), one of which (circled and enlarged) is a cluster of three subpiles: two collages of panorama pictures and a series of landscapes arranged by location, color, and saturation. This complex organizational scheme was not anticipated at the start, nor was it ever explicitly articulated – it emerged gradually through reflective interaction with the collection.

A third example illustrates this emergence even more strongly. Figure 6 shows another participant in mid-session. He has pulled out multiple sets of photos from an initial layout grid (within the marked rectangle) and placed them in piles (circled) wherever empty space was available — including the "holes" left from previous selections. His session proceeds in a series of steps, each driven by the accumulation of previous choices; the structure of his workspace at the end is determined by opportunistic, moment-to-moment revisions of the landscape of the developing space.

These and other close observations of participants organizing photos have been the basis for the design of Dynapad's region tools. Our goal is to automate the low-level

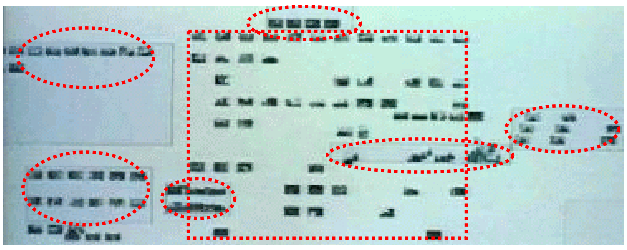

Figure 6: Opportunistic reassignment of space: structuring elements from initial grid (boxed) into piles (circled).

work involved in organization to provide more efficient and natural interaction.

2. Dynapad's Region-Tools

Let's begin the description of Dynapad's regional tools with the simplest, most literal interpretation of a pile: the self-adjusting "clumps" of Figure 7. As we discuss elsewhere [2], the piles of Mander et al. [17] are "closed" until actively browsed (e.g. "rippled through"), which limits the capacity for items and arrangements to *remind* users of information and opportunities [25, 16]. In contrast, Dynapad's "open" clumps require little or no effort to access individual items, and therefore support activities (such as collection-browsing) that require frequent access to potentially many different elements. The extra space required by such a strategy is mitigated by Dynapad's zooming, which provides an essentially infinite workspace.

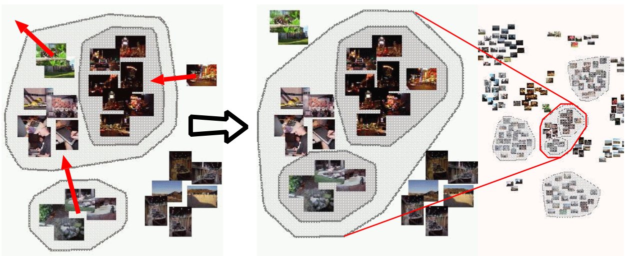

Figure 7: Dynapad's self-adjusting "clumps"

Dynapad's clumps, shown in Figure 7, automatically adjust their boundaries when images or sub-clumps are added or removed. In addition, when zoomed out these clumps "fuse" together: dragging any member will move the entire clump, preventing accidental disruption of its structure at small scales.

2.1. Implicit vs. Explicit Regions

In the examples of Figures 4-5, the piles are *implicit*: they exist only through the relative spatial density of their elements, and their meaning is only in the mind of the participant. In contrast, our those in Figure 7 (as well as Mander's proposed implementations [17]) are *explicit*: the spatial structures are reified as tangible, draggable objects with definite boundaries. Our goal is to provide "proactive" piles, whose behavior is a function of the pile's *meaning* or *role* within the workspace. One might imagine that such piles could be implemented equally well as explicit tools (as in Figure 7) or implicit regions (perhaps with a clustering algorithm). But such implicit piles are problematic because their "meaning" cannot "track" the pile if it moves.

An example of the problem is illustrated in Figure 8. Initially, group A occupies space S. The subject moves group A, then moves group B into A's former space (S). Should the "meaning" of pile A follow the objects, or remain in the space and apply to the new group B? Put another way: did pile S *move* or *change contents* (replacing A with B)? We cannot correctly interpret the action without knowing the user's intent.

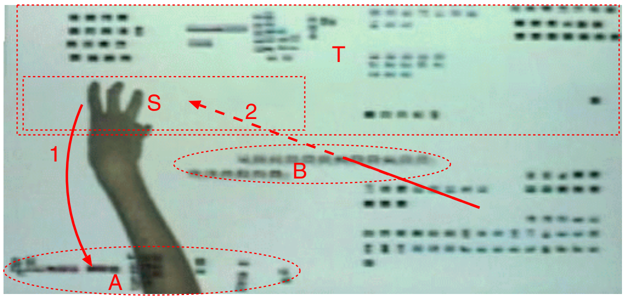

Figure 8: Ambiguous move of implicit piles

From the subject's self-reporting, we do know the correct interpretation: he has decided to use the top of the workspace (area T) as a timeline, and has arranged chronologically several event-specific piles. Group A, composed of "sunsets", was placed earlier and does not follow this scheme. Therefore he moves A out of the way to make room in the timeline for a new event, B. The former meaning ("sunsets") of space S *moves with the pile*, and that space is reclaimed by pile T to include sub-pile B, with its own meaning.

If the piles are *explicit*, like those of Figure 7, where the pile's behavior is bound to a tangible object, the user's decision to move the *pile itself* or just its contents disambiguates the interpretation. For this reason, although implicit versions are possible, all of Dynapad's region-tools are explicit, draggable objects. They reify meaningful groups of elements into explicit structures, remain open and accessible at multiple scales, are intuitive and nearly effortless to modify, and yet protect any substructure the user invests in them when they are moved, abutted, or even superimposed.

2.2. Beyond the Clump: Arrangement Tools

Despite useful structure-preserving affordances, the clumps of Figure 7 are a narrow literal interpretation of the "pile as specialized region" metaphor; beyond self-adjusting their size as elements are added and deleted, they offer no other *proactive* behaviors to assist the user in *creating* meaningful structure.

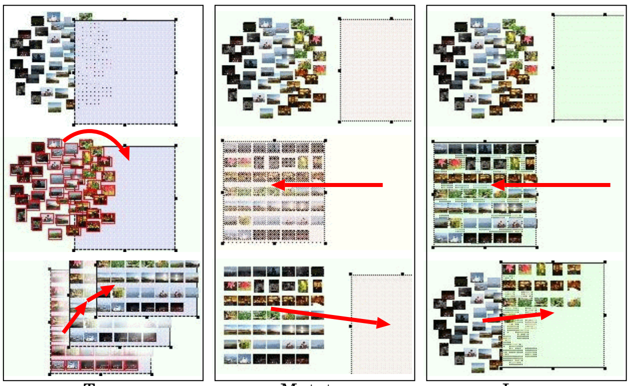

Figure 9: Arrangement tool examples: tray, mutator, and lens.

Figure 9 illustrates the behavior of three proactive region-tools: a *tray*, a *mutator*, and a *lens*. Each tool arranges a set of digital photos into a grid, sorted by the date and and time when the photo was taken[1]. This "gridding" and sorting automatically enriches a pile with a microstructure much like that which was manually invested in Figure 4: similar photos are aligned and adjoined by virtue of their similar timestamps, thereby aiding comparison. The *tray*, like the pile-tools above, behaves like a container which carries around (and rearranges) objects dropped into it. The *mutator*, in contrast, applies its effect wherever it's dropped, rearranging objects within its boundary but leaving them behind when moved. Finally, the *lens* leaves undisturbed the objects it's placed over, but instead arranges copies of them projected above it.

Another example of dynamically-enriched behavior is *linked brushing*: when the cursor rolls over any object, all other copies of it in the workspace are highlighted. This solves a common organizational challenge: needing to put an item into multiple categories. Items can simply be copied and placed in as many spatial roles as needed, and they remain visibly linked. We are experimenting with alternative linking representations to address, for example, cases where some linked instances are outside the viewing frame. We have also begun to support linked brushing not just between instances of the same object, but between objects related in various ways: for example, files in the same directory, photos taken the same day, papers by the same author, and citation relations. Any source of metadata could potentially be represented in this way. Representations of time, especially, have been shown to be effective in organizing and retrieving documents [8, 20], personal photographs [13, 10], and other media [19].

[1]The date information is extracted from the *exif* header embedded automatically in the image data by the camera.

Below we focus on both what the tools *share* and how they *differ*. Our system is designed to be flexible, and its architecture separates different aspects of the tools' behavior into independent components which can be combined in multiple ways. Although there are many alternatives to the current architecture, we believe that the separation we have adopted reflects important design dimensions.

2.3. Tool Effects

One component of a tool's behavior is its *effect*, what it does with the objects it acquires. The three examples above share the same effect component, ordering their contents by time in a grid. But we can imagine countless other operations on sets of images. An effect may impact both how an object is *positioned* (as with these examples) and how it is *displayed*. Many classical implementations of lenses [6, 9] are display-only: for example, highlighting objects of certain parameters, displaying labels, making corrections to text, and so on. We have only begun to explore the possibilities of displaying metadata and otherwise informative depictions of papers and other digital content. However, because of people's extensive use of space in organizing information, and because local spatial rearrangement presents unique challenges for lens-like tools, we have focused especially on arrangement effects like those in Figure 9.

A variant on grid-arrangement is a *timeline*. As illustrated in Figure 10, the timeline spacing can be adjusted to spread out objects for visibility (top panel) or align them precisely according to their timestamp, producing a histogram effect (middle panel). The bottom panel of Figure 10 shows how tools can be combined: a tightly-clumped timeline uses additional lenses to "pull apart" two of the clumps shown in the middle panel and show the temporal substructure of each. Even more than the time-grids, the timeline helps to categorize photos *by event*, a strategy employed frequently by our subjects (as in Figure 5, for example).

Both the timeline- and grid-arrangement effects automate several common arrangement subtasks (sorting, adjoining, and aligning) and the timeline also automates spacing. But these tools' ease of use costs them *flexibility*: they cannot be invested manually with the same richly-detailed substructure as basic clumps. As a compromise, we have anticipated developing another layout variant: the "gathering" clump. Basic clumps do no arrangement and force the user to manually adjoin, align, and de-occlude items. A "gathering" version (Figure 11) draws items into a tight grid or other pattern, like marbles in a rubber-band, providing automatic adjoining and aligning without imposing any particular sorting. A more automated variant could ensure two-dimensional partial sorting: items would retain (approximately) the topological ordering of a X-Y scatterplot, but condensed into a tight "gather-plot".

Figure 10: Timeline effects, top to bottom: Loosely-clumped, Precisely-spaced, Precise with additional timeline-lenses.

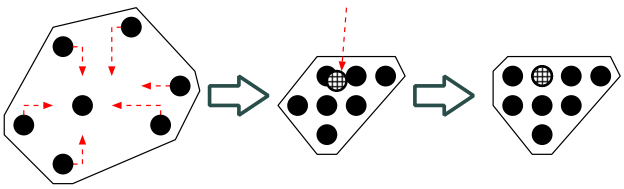

Figure 11: Proposed "gathering" clump

The arrangement examples so far have all used a timestamp, extracted from the *exif* data automatically encoded in an image by the camera. However, this timestamp is but one example of a *metadata parameter*: a function extracting some value from an object for use in various effects (like layout). For photos, other *exif*-based parameters might use camera focal length, for example, to distinguish close-ups from landscapes, or brightness to distinguish indoor and outdoor scenes. In general, parameters can utilize any source of "faceted metadata" [27] for photos or other content, and can be combined with existing effects– for example, a timeline can easily become a "nameline" where items are sorted and spaced alphabetically by filename or owner.

2.4. Operational "Syntax"

The differences between the tray, lens, and mutator examples illustrate another important design dimension. Although the tools have the same arranging-effect, they differ in the rules by which they acquire and retain their contents, which we might call the tool's "syntax".

Although the term "syntax" may seem counter-intuitive, there is a fruitful analogy between symbolic languages and Dynapad's spatial "language" of interaction: the spatial and temporal relations between components determines the "meaning" or effect of a construction. If we think of a tool's effect as an operation, then its syntax determines the set of operands. Such coordination of operation and operands is a design choice which underlies many direct-manipulation interfaces. Consider this simple example: in most text editors, copying text requires two actions, first selecting the text fragment, then specifying a copy operation via menu or keystroke. We might describe this as a temporally-serial "Object-Verb" syntax. Other examples employ a "Verb-Object" syntax, in which the user selects first an operation or tool and then its target. Still other interfaces allow the action and target to be specified concurrently: for example ToolGlass [6], ToolStone [21], and various gestural interfaces [12]. The effectiveness of these tools may be due in part to their more intuitive, holistic operational syntax, and we mean for Dynapad's region-tools to share this strength.

For implicit piles, the relation between a region and its members is merely spatial: a pile comprises those objects within its spatial boundary. However, when the pile is an explicit, movable object, its relation to its contents requires a more precise operational definition. The *history* of an arrangement may be important: for example, there may be a difference between moving an object into a region and moving a region over an object, or between moving items one at a time or as a batch.

While a tool's *function* determines the conditions and activities in which it is useful, it is a tool's syntax which determines its *affordances*, how you *invoke* its function, how you *interact* with it during use. We have implemented these dif-

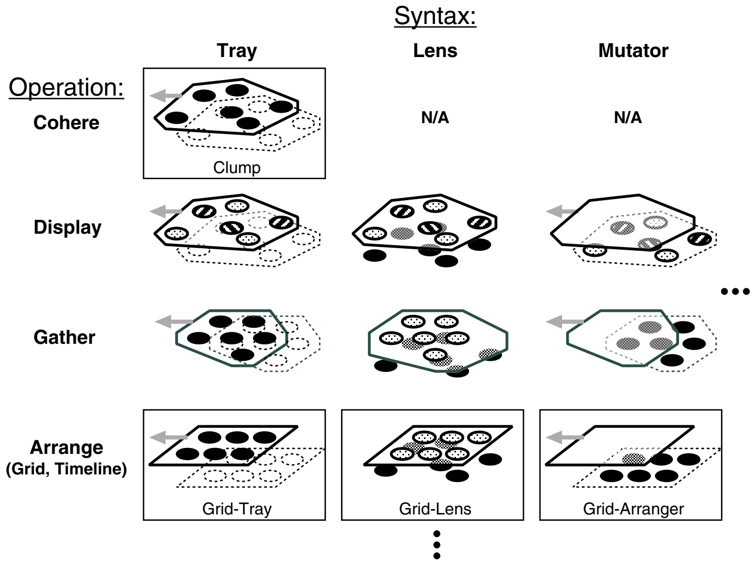

Figure 12: A small sample of the design space, including the tools detailed above.

ferent variants of syntax not just to search the design space for a single, consistently "best" design, but because we have observed these variants to produce qualitatively different usage strategies.

For example, one advantage of the "tray" syntax (being both retentive and non-absorptive) is that it allows users to reclaim space for additional uses – literally "stacking" multiple roles for the same space. But we have also observed hazards unique to Trays. For example, one subject used a tray as a sorted "staging" area, from which she drew items into other piles. The tray needed to be sufficiently large not to occlude its copious contents, but then the subject had to drag selections a considerable distance to clear its edges. Later, she decided to pre-sort the items with a mutator, then employed trays to overlap that space, making the task much easier.

2.5. Implementation Details

All of Dynapad's region tools share a common architecture based on a simple principle of "membership updates": regions act when objects enter, leave, or move within them.

Each region *effect* (e.g. timeline-arrangement) provides an interface invoked by callbacks on four different events:

- A moving region can *absorb* new members and *abandon* current members.
- A stationary region can *receive* objects moved to it or *release* members moved out of it.
- Either can *update* members which move within it.
- Finally, a region can perform a *finish* action once all membership changes have been made.

Although membership is discrete (in or out), mathematically continuous effects (e.g. a Gaussian lens) can be implemented by considering a member's exact position after an *update* and extending the boundary of the region to include the entire domain where the effect function is above some threshold.

A region's *syntax* specifies which of the events are active – for example, a tray does not *absorb* or *abandon* its members, while a lens does. Syntax also determines several other policies for maintaining region membership. We have found that a wide range of syntax variants may be specified with only five binary variables:

- An *absorptive* region may *absorb* and *abandon* objects.
- A *proxy* region (a lens, for example) applies all effects to copies of its members rather than the originals. This assures that it does not disrupt anything in the workspace or other regions, even if they share its members.
- A *greedy* region *receives* first and tries to receive all dropped objects (even if it does not contain them) whenever it encloses the cursor position, the drop location. This affords "funneling" a large selection into a small container or button, and is an intuitive default in many applications.
- A *retentive* region carries along all of its members (or proxies, if a lens) when moved.
- A *possessive* region does not share its members with any other possessive region; that is, objects cannot belong to more than one possessive region.

These properties of our three syntax examples are summarized in Table 1.

Property	Tray	Lens	Mut.
Absorptive (grabs underlying objects)		X	X
Proxy (operates on copies)		X	
Greedy (takes all moved objects)	X		
Retentive (carries members)	X	X	
Possessive (limits sharing)	X		

Table 1: Summary of Syntax Variables

2.5.1. Algorithm

Dynapad's actual membership-update algorithm is complex due to optimizations and implementation-specific details, but its worst-case scenario[2] can be expressed relatively simply. Whenever a set of objects Ω_m, which may include regions, are moved and "dropped" with the cursor at point *P*:

1. Find "damaged" region *D*, the spatial union of Ω_m before and after the move.
2. Find all objects ($\Omega \supset \Omega_m$) intersecting *D*, all regions ($R \subset \Omega$) intersecting *D*, and moved regions ($R_m = R \cap \Omega_m$).
3. *Moving objects:*
   For each region $r \in R$ (ordered top-down) [3] and each moved object $\omega \in \Omega_m, \omega \neq r$:
   (a) if r uses *proxies*, ω_r is a private copy of ω, else $\omega_r = \omega$;
   (b) $\omega_r \in r$ if r *contains*[4] ω,[5] or
   r *contains P* and r is *greedy*;
   (c) if $\omega_r \in r$ now but not before, r *receives* ω_r;
   if $\omega_r \in r$ both before and now, r *updates* ω_r;
   if $\omega_r \in r$ before but not now, r *releases* ω_r.
4. *Moving regions:*
   For each moved region $r \in R_m$ (ordered bottom-up) and each object $\omega \in \Omega, \omega \neq r$:
   (a) identify ω_r as before;
   (b) $\omega_r \in r$ if r *contains* ω;
   (c) if $\omega_r \in r$ now but not before, r *absorbs* ω_r;
   if $\omega_r \in r$ both before and now, r *updates* ω_r;
   if $\omega_r \in r$ before but not now, r *abandons* ω_r.
5. For each $r \in R$, r *finishes*.

[2]The complexity of the worst case is $O(n \times r)$, where n is the total number of objects (including regions) and $r \leq n$ is the total number of regions.

[3]As with gravity, a moved ω is *received* from the top and a stationary ω is *absorbed* from the bottom.

[4]In our current design, r *contains* ω if r encloses ω's center and ω does not enclose r.

[5]For lenses to be compositional, the output ω_r of one must be the input ω of another. Currently we do not permit this. To do so would require that any proxy ω_r is incrementally added to $\Omega - \Omega_m$ prior to step 4 above. For an exhaustive discussion of lens-composition challenges, see [9].

In summary, Figure 12 illustrates a small sample of the design space of Dynapad's region-tools, including the examples we've discussed. The basic "clump" has essentially no effect, beyond the cohesion resulting from its tray-like syntax. Traditional lenses and see-through or spatially-situated tools [6, 21] are included in the space as two forms (Lens, Mutator) of a display-only effect. The arrangement trays, lenses, and mutators described above are but a few examples of the vast space of possibilities. We emphasize that because the different components of the tools' behavior are independent and complementary, any new effect, parameter, or syntax created can be combined with existing components, thereby creating many tools with relatively little effort (for example, a "filename-grid-lens", "filename-line-tray", etc).

3. Regional Supportive Physics

The "physics" of a workspace are simply the ways objects behave and the affordances they offer users. Naturally, the ideal physics depend on the cognitive roles that space plays. Since different regions can play different cognitive roles in a larger activity, it makes sense that regions could benefit from different, role-specific physics. Many applications make use of this principle implicitly. For example, the ART system [18] is a spatially-based environment for composing text documents. It features three different sub-spaces which support three different aspects of composition, each with a physics appropriate to that sub-task. Other well-known examples include: (1) Windowing systems: dragging an icon from one folder (region) to another may have different effects (linking, copying, or moving) depending on the relationship between the source and destination regions. (2) Spreadsheets: different cells play different roles in a larger computation; the same number "dropped" in different places can have different effects.

The work we report here is an effort to generalize a notion of spatially-located physics — to develop an infrastructure of regional tools whose physics both automate the creation of microstructure and guide the management of macrostructure at multiple scales, to create variegated, interactive, task-specific workspaces. But we also realize that no software, however clever, will be able to anticipate the full variety of arrangements and strategies people employ in such reflective and opportunistic interactions. A tool to support, rather than dominate, such activity must afford the user both the authority to override its initiatives and the expressiveness to employ a variety of strategies. The unlocking and reediting of a portrait-collage as depicted in Figure 2 is one example. In short, any automation or physics must be adequately *humble*.

This philosophy also underlies the implementations of the pile metaphor proposed by Mander, Solomon, and Wong [17]. Another system kindred in spirit is Presto [7]. Dourish and colleagues implemented an experimental architecture for creating what they termed "fluid interactive document spaces." Presto provided an attribute-value-based structure to better support the richness of everyday document use. Vista, a Presto browser, gives direct manipulation access to the document space. They make use of piles as proposed earlier [17] but "have not yet implemented some of the richer features of the Apple piles prototypes, including 'messy' and 'neat' piles, as well as rippling through piles to search for documents" [7].

3.1. History Visualization

One goal of our research is to support interaction with digital collections over long time periods, during which arrangements and representations can change significantly. Therefore, we expect great benefit from the ability to revisit and visualize the long-term history of a workspace.

Several other systems have pioneered the use of time-based organizations, such as those in Lifestreams [8] and in Rekimoto's innovative TimeScape [20]. We can think of interacting with a workspace as *composing an artifact* [1] with a design history. Some research has addressed the challenge of archiving and recovering design history, notably the Designers' Outpost [15]. One challenge is that design history is often non-linear; designers reload saved versions and diverge from earlier changes, producing a tree-like topology of design states. The Designers' Outpost handles gracefully these complex histories, using branching timelines to represent diverging design threads.

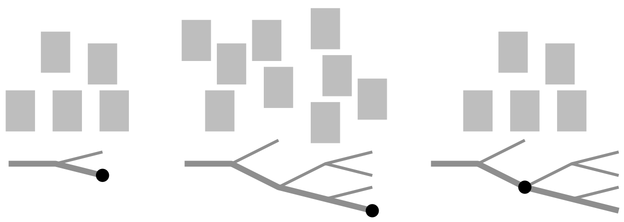

Figure 13: Historically-contextualized states with state-trees (simplified for clarity). The heavier branch shows the "active" path; the dot is the "present" state. *Left:* Original state **a**; *Center:* Later state **b**; *Right:* Restored state **a**.

Similarly, Dynapad logs all changes in a workspace and maintains a history tree of *all past states*, even if not explicitly saved. This is so that the workspace-artifact has an accessible analogue of all phases of the user's experience with their collection. Figure 13 shows abstract characterizations of two different states of a workspace situated in a branching history: state **a** is followed eventually (after some backtracking) by state **b**, then **a** is restored. The "present" workspace returns to its original state **a**, but the historical context, represented by the tree, now stores additional "futures". Users can backtrack and advance stepwise along the "active" path by repeatedly "undoing" and "redoing", or jump to any point by clicking on the state-tree. This logging mechanism is also a source of ethnographic data to support more thorough analysis of the fine-grained structure of users' behavior.

4. Conclusions

This paper describes the development of spatial tools for managing personal information collections. Our primary research objective is to explore and generalize the notion of a "pile" as the foundation for a versatile suite of region-tools which provide unobtrusive assistance for organizational and other sensemaking activities. In summary, we distill our conclusions into five principles:

1. Interaction with personal digital collections is an activity which increasingly demands better support from software environments, and serves as an ideal domain to study and apply the capacities of multiscale image-based workspaces.
2. Collection management in particular elicits variegated uses of space, where the flow of activity could be facilitated by automating common tasks in specialized areas, but requires that the automation remain unobtrusive and highly flexible.
3. Since 'piling' strategies are ubiquitous, intuitive, and effective, variations of the "pile" can offer a natural metaphor for interaction with "region-tools" which provide such automation.
4. The design space of Dynapad's region-tools can be characterized in a relatively minimal interface of four callback events and five binary properties, combinations of which correspond to several intuitive interaction metaphors.
5. There is a natural and productive dissociation between two facets of a region-tool's behavior: its *effect*, the operations that it performs on digital objects – and its *operational syntax*, how we interact with it. In our experience as developers, this dissociation permits our toolkit to be extended with relative ease to a variety of functionality and interactive styles. And from our observations of users, we are encouraged by the potential for this design framework to improve the effectiveness of region-tools.

5. Acknowledgments

This work has been supported by National Science Foundation grants #0113892 and #9873156 to Jim Hollan. We also gratefully acknowledge gift support from Mitsubishi Cambridge Research Lab and from Intel Corporation. Pierre Fastrez was supported by a Postdoctoral Fellowship from the University of Louvain, Belgium, a Fulbright fellowship, and a B.A.E.F. Honorary Fellowship.

We are fortunate to benefit from the collective efforts of the other members of the current Dynapad development effort: Ron Hightower, Ron Stanonik, Dana Dahlstrom, and Shaun Haber. As always, we appreciate comments and suggestions from the other members of the DCOG-HCI lab.

References

[1] D. Bauer. A multiscale workspace for managing and exploring personal digital libraries. In *ACM UIST'03:Doctoral Symposium*, November 2003.

[2] D. Bauer, P. Fastrez, and J. Hollan. Computationally-enriched "piles" for managing digital photo collections. In *2004 IEEE CS International Symposium on Visual Languages and Human-Centric Computing (VL/HCC 2004)*. IEEE Computer Society, 2004. (in press).

[3] B. B. Bederson. Photomesa: a zoomable image browser using quantum treemaps and bubblemaps. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*, pages 71–80, 2001.

[4] B. B. Bederson and J. D. Hollan. Pad++: A zooming graphical interface for exploring alternate interface physics. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 17–26, 1994.

[5] B. B. Bederson, J. D. Hollan, K. Perlin, J. Meyer, D. Bacon, and G. Furnas. Pad++: A zoomable graphical sketchpad for exploring alternate interface physics. *Journal of Visual Languages and Computing*, 7:3–31, 1996.

[6] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. Toolglass and magic lenses: the see-through interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 73–80. ACM Press, 1993.

[7] P. Dourish, W. K. Edwards, A. Lamarca, and M. Salisbury. Presto: An experimental architecture for fluid interactive document spaces. *ACM Transactions on Computer-Human Interaction*, 6(2):133–161, 1999.

[8] S. Fertig, E. Freeman, and D. Gelernter. Lifestreams: an alternative to the desktop metaphor. In *Proceedings of ACM SIGCHI Conference*, pages 410–411, 1996.

[9] D. Fox. Composing magic lenses. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 519–525. ACM Press/Addison-Wesley Publishing Co., 1998.

[10] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Proceedings of the second ACM/IEEE-CS joint conference on Digital libraries*, pages 326–335. ACM Press, 2002.

[11] K. D. Grant, A. Graham, T. Nguyen, A. Paepcke, and T. Winograd. Beyond the shoe box: Foundations for flexibly organizing photographs on a computer. Technical report, Computer Science Department, Stanford University, 2003.

[12] F. Guimbretiere and T. Winograd. Flowmenu: combining command, text, and data entry. In *Proceedings of the 13th annual ACM symposium on User Interface Software and Technology*, pages 213–216, 2000.

[13] H. Kang and B. Shneiderman. Visualizing methods for personal photocollections: Browsing and searching in the photofinder. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME2000)*, pages 15390–1542, 2000.

[14] D. Kirsh. The intelligent use of space. *Artificial Intelligence(1-2)*, 73:31–68, 1995.

[15] S. R. Klemmer, M. Thomsen, E. Phelps-Goodman, R. Lee, and J. A. Landay. Where do web sites come from?: capturing and interacting with design history. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1–8. ACM Press, 2002.

[16] T. Malone. How do people organize their desks?: Implications for the design of office information systems. *ACM Trans. on Office Information Systems*, 1(1):99–112, 1983.

[17] R. Mander, G. Salomon, and Y. Y. Wong. A "pile" metaphor for supporting casual organization of information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 627–634. ACM Press, 1992.

[18] K. Nakakoji, Y. Yamamoto, S. Takada, and B. Reeves. Two-dimensional spatial positioning as a means of reflection in design. In *Proceedings of ACM DIS'00*, pages 145–154, 2000.

[19] R. Pea, M. Mills, J. Rosen, K. Dauber, W. Effelsberg, and E. Hoffert. The diver project: Interactive digital video repurposing. *IEEE Multimedia*, 11:54–61, 2004.

[20] J. Rekimoto. Timescape: A time-machine for the desktop environment. In *Proceedings of ACM CHI'99 Extended Abstracts*, pages 180–181, 1999.

[21] J. Rekimoto and E. Sciammarella. Toolstone: effective use of the physical manipulation vocabularies of input devices. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 109–117. ACM Press, 2000.

[22] G. Robertson, M. Czerwinski, K. Larson, D. C. Robbins, D. Thiel, and M. van Dantzich. Data mountain: Using spatial memory for document management. In *Proceedings of ACM UIST'98*, pages 153–162, 1998.

[23] K. Rodden and K. Wood. How do people manage their digital photographs? In *Proceedings of ACM CHI'03*, pages 409–416, April 2003.

[24] A. Sellen and R. Harper. *The Myth of the Paperless Office*. MIT Press, 2001.

[25] S. Whittaker and J. Hirschberg. The character, value, and management of personal papaer archives. *ACM Transactions on Computer-Human Interaction*, 8(2):150–170, 2001.

[26] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrsion, and P. Pirolli. Using thumbnails to search the web. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 198–205. ACM Press, 2001.

[27] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the conference on Human factors in computing systems*, pages 401–408. ACM Press, 2003.