

Characterizing users' visual analytic activity for insight provenance

David Gotz* and
Michelle X. Zhou

IBM Research, 19 Skyline Dr, 1N-C13,
Hawthorne, New York 10532, USA.

*Corresponding author.
E-mail: dgotz@us.ibm.com

Abstract Insight provenance – a historical record of the process and rationale by which an insight is derived – is an essential requirement in many visual analytics applications. Although work in this area has relied on either manually recorded provenance (for example, user notes) or automatically recorded event-based insight provenance (for example, clicks, drags and key-presses), both approaches have fundamental limitations. Our aim is to develop a new approach that combines the benefits of both approaches while avoiding their deficiencies. Toward this goal, we characterize users' visual analytic activity at multiple levels of granularity. Moreover, we identify a critical level of abstraction, *Actions*, that can be used to represent visual analytic activity with a set of general but semantically meaningful behavior types. In turn, the action types can be used as the semantic building blocks for insight provenance. We present a catalog of common actions identified through observations of several different visual analytic systems. In addition, we define a taxonomy to categorize actions into three major classes based on their semantic intent. The concept of actions has been integrated into our lab's prototype visual analytic system, HARVEST, as the basis for its insight provenance capabilities. *Information Visualization* (2009) **8**, 42–55. doi:10.1057/ivs.2008.31

Keywords: analytic activity; insight provenance; visual analytics

Introduction

Visual analytic systems have enabled analysts to interactively explore and derive insights from large corpora of information by exploiting human visual perception and abstract reasoning.¹ In many applications, it is essential to track an insight's provenance to record how and from where each insight was obtained. We use the term *insight provenance* to refer to a historical record of the process and rationale by which an insight is derived during a visual analytic task. Insight provenance includes the user's relevant visual activity, the information being explored as well as the derived insight itself.

A number of existing visual analytic systems have been equipped with tools specifically designed to capture insight provenance.^{2–7} Although these tools vary widely, they can be broadly classified into two categories based on the level of user involvement: manual capture of insight provenance and automatic capture of insight provenance.

The manual approach to insight provenance requires users to explicitly record their visual analytic activity (including their notes or argument structures) to document their findings.^{3,4,7} This approach is often very effective for capturing the high-level rationale by which users connect individual insights into an overall conclusion. Because users manually record their own logic, this approach accurately documents the semantics of a user's insight provenance. However, the manual approach has limitations.

Received: 7 November 2008
Accepted: 16 November 2008



Most critically, it is difficult to scale up the manual approach to handle the highly interactive nature of visual analytic activity. During a visual analytic task, users typically perform a very large number of activities at a very fast pace. Each of these activities – every query, filter or sort request – is motivated by a logical rationale. However, it is too laborious and impractical for a user to manually record each individual activity and its rationale due to the overwhelming amount of information involved.

For this reason, users often record just the final state of a visualization and tag it with a high-level description. Users typically omit from their documentation the intermediate steps that led to the insight. Moreover, users often omit seemingly unimportant visualizations from their notes even if they directly motivated additional lines of inquiry. As a result, critical information may be lost and the manual approach fails to capture a user's insight provenance comprehensively.

For example, a financial analyst who has been looking through stock market data would typically capture only a select number of visualization states that support his/her final investment recommendations. The details of the analyst's exploration path, as well as his/her intermediate findings, would be lost. The analyst's client would be unable to verify exactly how the investment recommendations were derived.

Given the limitations described above, systems that automatically capture insight provenance^{2,5,6} are a compelling alternative. Unlike the manual approach, a system that automatically records a user's analysis process could potentially capture a comprehensive model of insight provenance. However, the automatic approach has its own limitations.

In particular, most existing visual analytic systems are event-based systems that are designed to recognize and process specific, often low-level user interaction events like mouse clicks and drags, but can rarely understand and capture the semantics of such events (for example, the analytic purpose of a user's mouse drag). In addition, during visual analysis the high rate of user activity often creates a large number of low-level user interaction events that grows enormously as the analysis unfolds.

As a result, it is extremely difficult for these systems to organize the large linear list of user interaction events into semantically meaningful segments of activity. It is even more challenging for such a system to infer the high-level semantic constructs that can capture the complex, nonlinear nature of a user's visual analysis process.

To capture insight provenance more effectively, we are investigating a new approach that combines the benefits of both manual and automatic approaches while addressing their limitations. Our approach allows for the automatic capture of both a semantic and comprehensive record of user activity, from which we can infer the high-level logical constructs of a user's analysis with minimal user involvement.

Our work is based on the hypothesis that we can achieve this goal by developing a semantics-based rather than

event-based visual analytic system. Instead of reacting to low-level user interaction events (for example, clicks and drags), such a system is designed to support and recognize a set of semantic but general user activities (for example, query and filter) that are broadly applicable across a variety of different visualization tools and applications. The set of user activities can then be used as semantic building blocks from which the system can automatically infer the higher-level logical structures within a user's analytic activity for use as insight provenance.

To identify an effective set of semantic building blocks, we characterize a user's visual analytic behavior at multiple levels of granularity based on the semantic richness of user activity. Motivated by Activity Theory⁸ and our own empirical studies of analysts' behavior,⁹ we model user visual analytic activity at four levels: tasks, sub-tasks, actions and events. *Tasks* and *sub-tasks* represent high-level, logical structures of a user's analytic process, such as the user's cognitive goals and sub-goals. *Tasks* and sub-tasks have rich semantics and are often domain or application specific (for example, the task of investigating the financial market for investment recommendations). The *action* tier represents the individual *executable* semantic steps, such as making a data inquiry, taken by a user while working toward their analytic goal. At the bottom of our model, *events* correspond to the lowest-level of user interaction events (for example, a mouse click or a menu item selection) which carry very little semantic meaning.

In this paper, we expand on our paper first published at IEEE VAST 2008.¹⁰ We focus on the action tier, which uniquely bridges the gap between higher-level logical constructs and the lowest-level user interaction events. Because of its unique role, we use the action tier to identify the semantic building blocks in our approach to insight provenance. In particular, we characterize the action tier by a number of dimensions, such as the type, intent and parameters of each user action. Based on this characterization, we develop an action taxonomy that identifies a core set of semantic building blocks for our approach to insight provenance. In addition, our characterization can be used to guide the future definition of new actions to expand the set of building blocks as required by new functionality.

Related Work

Our multi-tier characterization of user visual analytic activity is based on Activity Theory, which describes how tools act as mediators of human behavior.¹¹ Activity Theory has been widely used by the HCI community to help explain how human users interact with computerized tools.⁸ Directly motivating our work, it includes a three-level hierarchy that characterizes human activity at multiple levels of abstraction, ranging from high-level task motivations to low-level human operations. Adopting this multi-tier activity model, we specifically

tailor the model to characterize users' visual analytic behavior for insight provenance.

Our work is also influenced by previous empirical studies that examine users' visual analytic behavior. These include studies that capture and analyze the high-level structure of user visual analytic behavior,^{12–14} and others that focus on examining lower-level user analytic activity.^{15,16} These studies directly motivate our layered approach to insight provenance, which captures user activity comprehensively at multiple levels of granularity.

Also related are a number of visual analytic systems that provide tools for preserving insight provenance, an important aspect of visual analytics.^{1,17} While each system takes a unique approach, they can be classified into two broad categories: (1) manual capture of insight provenance and (2) automatic capture of insight provenance.

Manual approaches include user-created notes,³ manually authored diagrams illustrating a user's analytic steps⁴ and user-built structured argumentation graphs.⁷ As these tools have users themselves manually record their own logic, the captured insight provenance provides a clear, logical view of a user's analytic process. Nevertheless, it is often too onerous for individual analysts to manually record every step that leads them to their insights. As a result, key details of a user's insight provenance may be lost.

In contrast, automatic approaches attempt to systematically capture the full history of a user's analytic process. For example, the VisTrails system² captures chains of visualization operations that represent scientific visualization workflows. There are also visual analytic systems that record histories of user visual operations and the parameters of these operations.^{5,6} Although these tools comprehensively and faithfully record user analytic activity, they cannot abstract the high-level semantic constructs obtained in the manual approach.

Recognizing the limitations of each approach, Groth and Streefkerk¹⁸ introduced a state-based interaction model which combines both automatically captured visualization states with manually authored annotations that capture a user's fine-grained rationale for each individual interaction. However, the annotation process faces the same limits as other manual-based methods and can be onerous for complex tasks.

Our goal is to combine the benefits of both manual and automatic approaches while avoiding their deficiencies. Toward this goal, our work presented here is to characterize user visual analytic activity at multiple levels of granularity. More importantly, we identify the right level of abstraction that can be used to represent a set of general but meaningful user activity primitives. In turn these activities can be used as semantic building blocks for a visual analytic system to understand and infer the meaning and rationale of such behavior.

Most similar to our approach is the Aruvi system.¹⁹ Aruvi supports two ways of preserving a user's insight provenance. First, it automatically records a user's navigational steps. Second, it provides an interface

component that allows users to manually add notes. However, the granularity of a user's navigational steps are determined by application-specific heuristics (for example, when the mouse pointer exits from a particular UI panel) that do not generalize for use with other visual metaphors or applications. In contrast, our action-tier focuses on abstracting meaningful user actions that are general across a range of visualization tools and domains.

In other work, a number of taxonomies have been developed that characterize visualization activity. Broadly speaking, these efforts fall into two distinct families: system-oriented taxonomies and user-oriented taxonomies.

System-oriented taxonomies focus on describing visualization or data operations.^{20–24} For example, Chuah and Roth²² define a set of operators, called *basic visualization interactions* (BVIs), describing various data and visual interaction operations (for example, *set-graphical-value*). Because these taxonomies are not designed to model user activity, they are most applicable for representing a system's response to user activity, not the user activity itself.

In contrast, user-oriented taxonomies typically characterize high-level human cognitive tasks or visual perceptual behaviors.^{25–28} For example, Amar *et al*²⁵ have come up with 10 basic task types describing various users' needs in information analysis. Existing user-oriented taxonomies often characterize behavior at granularities which correspond closely to the Task or Sub-task tiers in our characterization. Compared to the previous taxonomies, our work is unique in identifying the Action tier, a granularity of activity that bridges the semantic gap between high-level human cognitive activity and low-level user visual interaction events.

Observations of User Visual Analytic Activity

Visual analytics is a complex process. Any useful model of insight provenance must accurately capture its nonlinear and progressive nature. To better understand and characterize a user's visual analytic behavior, we invited 30 users and asked them to perform typical visual analytic tasks using commercial visualization tools.⁹ We share the key insights obtained from our observations and discuss how they motivate our work.

Tasks and tools

We designed two real-world analysis tasks based on common business-oriented activities. The first task asked users to analyze stock market data for investment opportunities, focusing on the financial and technology sectors. The second task had users research business travel facilities for a complex multi-city event, asking them to produce several competing alternatives (for example, meeting places and hotels). To avoid potential biases (for example,



unrealistic data or unreliable tools), we provided our participants with commercial-grade visualization systems and real-world data to accomplish the two tasks. For the first task, participants were given access to the Map of the Market visualization tool from SmartMoney.com.²⁹ They were also allowed to use directly linked web resources such as analyst reports and related news articles. For the second task, participants were given access to a map-based travel tool that combines lodging, dining and entertainment reviews with corporate databases and policies regarding hotels, rental cars and office locations.

Each user was allotted 1 hour to accomplish his/her task. During this process, each user's activity was both video taped and manually recorded by a human moderator. Participants were also given proper tools to record hand-written notes to capture visual snapshots of their screens (for example, pens, paper and the screen capturing tool *SnagIt*).

Observations

We analyzed the extensive records of user activity captured in our experiments (for example, videos, user notes and the moderator's logs) and distilled three key observations. These observations have helped guide us in our characterization of visual analytic activity.

Multiple tiers of analytic behavior

All participants demonstrated distinct hierarchical approaches to their tasks. In all cases, there was a clear pattern of task decomposition that divided each high-level task into smaller and more concrete sub-tasks. For example, when performing the market analysis task, one participant began by uncovering the best and worst performing stocks in financial and technology sectors, respectively.

From our observations and conversations with the participants, several people consciously took a hierarchical approach to the task as part of a concrete plan. For others, the decomposition was less formal but implicitly done in *ad hoc* fashion over the course of the task.

Similarly, we observed the decomposition of sub-tasks into individual user actions that could be performed with the provided tools. Each action represented a distinct user intention to elicit a response from the visual tool. For example, during the market analysis task, a common sub-task was to identify the best-performing companies in a particular sector. To accomplish this sub-task, one user carried out the following actions:

- Zoom in to the technology sector to view stock performance of various industries in the sector.
- Create notes to record the strongest industries.
- Zoom in further to the software industry to examine stock performance of various companies in the industry.
- Create another note for the top software performers.
- Change the view to show just the past 26 weeks.

- Record notes on top software performers at this range.
- Several changes to the time range of data being viewed.
- Zoom out to the entire technology sector.
- Zoom in to focus in on the internet industry.
- Create notes on top internet performers.

Our observations are consistent with the hierarchical nature of human activity predicted by Activity Theory.⁸ In particular, we observed that a user's visual analysis process is inherently hierarchical, including high-level user cognitive activities and lower-level, concrete user actions (for example, creating a note to annotate his/her discovery).

A common set of semantic building blocks

The participants in our experiments performed two very different tasks using two very different visualization tools. One group of users was investigating investment opportunities, whereas the other was planning a complex business event. Yet at a particular granularity, we found a common set of user actions that were repeatedly performed regardless of the visualization tool being used or the specific tasks being performed. For example, the user actions listed above (for example, *Zoom in* and *Create notes*) were performed by all of the participants in both tasks.

Moreover, our observations showed that such actions were motivated by a specific user purpose. For example, the intention of a user that *Zooms in* is to change his/her existing view of a set of information to display a larger or more detailed view of a particular region. The user intent for each action was the same across both tasks and tools, regardless when or where it was performed.

This finding supports our hypothesis that we can identify a common set of meaningful user actions on which one can build a semantics-based visual analytic system. This set of user actions can then serve as the set of semantic building blocks needed to automatically capture comprehensive insight provenance from which higher-level user intentions can be inferred.

Inferring logical provenance from user actions

Our third key observation is that users aggregate chains of actions on their way to discovering individual insights. We observed from our study that, at the action level, the insight discovery process typically contains two phases. First, in an *exploration phase*, a participant gathers or filters data, changes views or otherwise explores through the available information. Then, when the user has identified an interesting insight, the participant engages in an *insight phase* during which they document their discovery by taking notes, capturing a visualization snapshot, or both.

Since a user often explicitly signals the discovery of an insight via actions like note taking, a visual analytic system can use these cues to punctuate the sequence of user performed actions, marking important semantic boundaries in the recorded history of user activity. These

boundaries can then be used to infer how chains of user actions aggregate to form the provenance of an individual insight.

Multi-Tier Characterization of User Analytic Activity

Motivated by Activity Theory,⁸ we characterize user analytic behavior at multiple levels of granularity based on the semantic richness of the activity. In particular, we characterize user visual analytic activity at four levels: tasks, sub-tasks, actions and events. The top three tiers describe activities that inherently possess some degree of semantics, with tasks as the richest of all tiers. The bottom tier represents the lowest-level user interaction events, such as mouse clicks and key presses, which in isolation hold little semantic value (see Figure 1).

Four tiers

Throughout this section, we use the following concrete example scenario to help explain our characterization. Consider a stock market analyst, Bob, who is asked to use visualization software to analyze the stock market in order to make investment recommendations in two market sectors: finance and technology.

Tasks

The *task* tier captures a user's high-level analytic goals. These goals, which are often open ended or ambiguous, are what drive a user's overall analysis process. In practice, analysts often juggle multiple tasks at the same time. In the example scenario, Bob has just a single task T_1 :

- T_1 : Identify key market insights to generate investment recommendations.

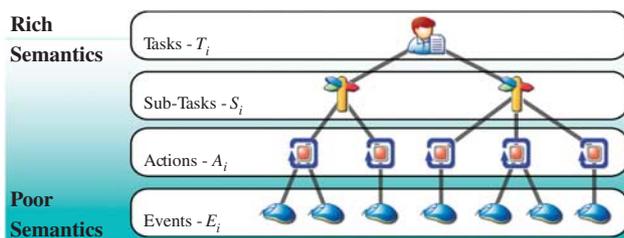


Figure 1: We characterize user analytic behavior at multiple levels of granularity based on the semantic richness of the activity. We define four tiers – *tasks*, *sub-tasks*, *actions* and *events* – which capture user behavior at decreasing semantic levels. Activities from each tier are combined into sequences to perform a single activity from the next highest layer as illustrated by the tree-like structure in the figure.

Typically, tasks are tightly coupled to the domain or application in which the user is working. For example, while T_1 is very appropriate for an investment analyst working with financial tools, a travel agent (working with a set of travel and transportation tools) would not likely perform the same task. While tasks vary widely, researchers have developed general taxonomies to help characterize their different types.²⁶

Sub-tasks

The *sub-task* tier corresponds to more objective, concrete analytic goals. For instance, sub-tasks (noted S_i) for T_1 in our example scenario might include:

- S_1 : Characterize the overall 52-week market trend in the technology sector.
- S_2 : Identify the best and worst performing financial companies over the last 8 weeks.

Analysts typically follow a divide-and-conquer approach, performing several sub-tasks to achieve the requirements of a single top-level task. As in the task tier, sub-tasks are often tightly coupled to the domain or application in which the user is working. For example, S_2 is a reasonable sub-task for the investment analyst in our example scenario. However, users in other domains would not likely encounter the same sub-task. Similar to tasks, sub-tasks can also be characterized using existing task taxonomies.²⁵

Actions

The third tier in our characterization of user analytic activity is the *action* tier. Here we define an *action* as an atomic analytic step performed by a user with a visual analytic system. Sequences of actions (noted as A_i) can be aggregated to accomplish individual sub-tasks. In our example scenario, Bob might start sub-task S_2 with the following actions:

- A_1 : *Query* for 8 weeks worth of stock market data.
- A_2 : *Split* companies by sector.
- A_3 : *Filter* to show only the financial sector.
- A_4 : *Sort* companies by their changes in stock price.
- A_5 : *Inspect* the company with greatest change in stock price to ask for more details such as financial reports.

Unlike the task or sub-task tier, where activity is typically domain or application specific, the action tier is generic. It represents a typical set of user actions that can be performed across different visualization tools and domains. For example, the *Query* action in A_1 can be performed by both financial analysts using stock market visualization tools, and travel agents using their own visualization tools. This contrasts strongly with both T_1 and S_2 , both of which were only appropriate in the investment domain.



While application independent, each action retains a rich level of semantics not found in low-level user interaction events, such as a mouse click or drag. For example, an isolated click event has little meaning without additional context. In contrast, an action such as *Query* or *Zoom* represents a semantically meaningful user activity. Actions, therefore, are unique in that they are both generic with respect to domains and tools, yet semantically rich in terms of user intent.

While taxonomies have been developed for both tasks and sub-tasks, relatively little attention has been focused on characterizing actions. However, actions are critical to our approach to capturing insight provenance since they represent the most basic, meaningful user analytic steps from which we can infer higher-level user tasks and sub-tasks. For this reason, we develop our own action taxonomy later in this paper.

Events

The fourth and lowest tier in our characterization is the *event* tier. An *event* E_i represents a low-level user interaction event which, in isolation, has little semantic value. This is in contrast to the three previous tiers (tasks, sub-tasks and actions) all of which had rich semantic connotations. Returning to the example scenario, action A_4 might consist of several low-level user events:

- E_1 : *mouse-drag* to select all companies to be sorted.
- E_2 : *mouse-right-click* to select open a popup menu.
- E_3 : *menu-select* to choose 'Sort' from a list of menu options.
- $E_3 \dots E_8$: *keyboard-events* to set sorting parameters (for example, choosing 'price change' as the property to sort and the selecting the sorting order).
- E_9 : *mouse-click* to submit the entered sorting parameters.

Discussion

Our model describes four distinct tiers of user behavior. However, real-world tasks can be very complex and are often sub-divided by analysts into far more than four levels of granularity. Specifically, analysts may decompose difficult problems multiple times within the task tier before arriving at a sub-task that is objective and concrete.

Therefore, the task tier in our model is intended to capture a range of behavior. Our model should not be interpreted to imply that any task, no matter how complex, can be subdivided directly into sub-tasks in a single decomposition. In practice, complex tasks are often sub-divided into smaller yet still complex pieces, many of which may remain open-ended and ambiguous tasks in their own right.

For example, consider the task T_1 defined in the previous section. This task can be decomposed into

individual tasks for each sector of interest: one to characterize the finance sector, another to characterize the technology sector. This behavior can be observed both in managers who delegate portions of a task to various colleagues, or in the behavior of a single analyst who is systematically solving a complex task one piece at a time.

Characterization of the Action Tier

The action tier is unique in our characterization of visual analytic activity because it is the only tier to combine two key properties. First, actions are domain independent, making them applicable to a broad range of user analytic tasks and applications. Second, actions represent meaningful units of user activity with well-defined semantics. Therefore, if a visual analytic system is designed to explicitly support user behavior in terms of actions, it can easily capture a semantic record of a user's activity without using domain knowledge or tool-specific heuristics based on low-level interaction events as often done in other systems.¹⁹

In this section, we first define a user action representation that captures its main properties. Then, based on that representation, we develop an action taxonomy that categorizes a set of common actions identified during our observations of visual analytic activity. Finally, we show how this taxonomy is used to define a representation that we use to semi-automatically infer how actions combine to satisfy higher-level sub-tasks.

Action representation

An action corresponds to an atomic and semantic analytic step performed by a user with a visual analytic system. A representation must therefore capture both (1) the general semantic properties of an action, and (2) the parameters that specify the particulars of an individual action instance. To satisfy this criteria, we represent an action with the following tuple:

$$Action = \langle Type, Intent, Parameters \rangle \quad (1)$$

The *Type* attribute defines a unique semantic signature for each action. For example, *Zoom* and *Pan*, which have unique semantic connotations, are both action types. *Intent* represents the primary user intention for invoking the specific action type. For example, both *Zoom* and *Pan* actions correspond to users' intention to change the visual presentation of data. Finally, an action's *Parameters* define the functional scope of an action and include the values required by a system to implement the action. For example, a *Query* action has a set of parameters that enumerates the main data concepts and data constraints that are required to formulate executable query statements (for example, SQL).

Our action representation is invariant to the underlying visual metaphor which supports it. For example, users can

Zoom in with a timeline just as they *Zoom* in on a map. In both cases, the user's action has the same intent and type. Therefore, both *Zooms* are considered semantically equivalent.

Similarly, our representation is independent of the underlying interaction events used to initiate an action. For example, Google Maps provides three distinct user interaction techniques for zooming in. Users can either (1) double click on the map, (2) drag a zoom slider or (3) click on a "plus" button. All three gestures are semantically equivalent and can be represented by the same action.

Catalog of actions

Using our action representation, we surveyed several visual analytic systems (both commercial products and our own prototype applications) and developed a catalog of common actions which we use to build our action taxonomy. In compiling this catalog, we reviewed systems from a wide variety of task domains (for example, finance, travel, real estate, import/export trade, text document analysis) and with a broad spectrum of visual metaphors (for example, statistical graphs, maps, timelines, treemaps). We also reviewed the detailed activity logs captured during our 30-person experiments.

Our analysis identified 20 distinct action types as summarized in Table 1. For each action, the table includes a formal definition (type, intent and parameters) as well as a brief description. Each action is described using one or more intents based on the primary user motivation. We use four distinct intents: (1) *data change*, (2) *visual change*, (3) *notes change* and (4) *history change*.

The list of actions represents a union of the actions we identified across all of the systems in our review. No single system supported the entire set of actions. Moreover, the action catalog is not comprehensive enough to represent every potentially possible action in systems beyond our survey. However, we believe that the catalog captures the most common user actions supported by a range of visual analytic tools.

The actions in this catalog serve two purposes. First, we use the action definitions to motivate the structure of an action taxonomy. Second, the list of actions types provides a common vocabulary for describing the basic semantic functions within a visual analytic system. For example, we use a subset of this catalog within our lab's prototype visual analytic system, HARVEST, to enable its semi-automated insight provenance functionality.

Action taxonomy

Using the 20 actions identified in our catalog, we further characterize the action tier to develop a general taxonomy. This taxonomy is designed for two purposes. First, its categories are used as the basis for our approach to inferring higher-level sub-tasks from a sequence of user performed

actions. In addition, the taxonomy serves as a guideline for others to expand the set of actions within our characterization.

Using *intent* a primary feature, we identify three broad classes of actions: *exploration* actions, *insight* actions and *meta* actions. Within each class, we further group actions into sub-categories based on their semantics (Figure 2). Next, we describe each action category and use concrete examples to elaborate key differences.

Exploration actions

Exploration actions are performed as users access and explore data in search of new insights. In terms of *Intent*, we classify all actions with a *data change* and/or *visual change* intention as exploration actions. Most actions fall into this category, including *Query*, *Filter*, *Pan* and *Zoom*. We further categorize exploration actions into two sub-categories: *visual exploration actions* and *data exploration actions*.

Visual exploration actions, such as *Sort* and *Change-Metaphor*, are actions where the primary user intention is to change the visual presentation of information that is already being presented to the user. For this reason, exploration actions whose only intent is *visual change* fall into this category. For example, the *Sort* action is primarily a request by the user to view the data elements in a data set (for example, shipments) in a certain order (for example, ascending order) by a particular data dimension (for example, shipment arrival date). Similarly, the *Change-Metaphor* action is primarily intended by a user to change the view of currently presented information (for example, a map-based display) to an alternative view (for example, a timeline display).

In contrast, data exploration actions are those in which the user intends to change the current data that is under investigation. Any exploration action with a *data change* intent falls into this category, such as traditional data inquiry behaviors *Query* and *Filter*.

Our classification of data and visual exploration actions is based on the user's primary intention, not the eventual system response. In practice, exploration actions in both categories can result in a system response that will modify both the data being viewed and the visual presentation of the data. To illustrate this point, we describe two exploration actions in detail: *Query* and *Split*.

Query: A *Query* action represents a user's request to bring a specific data set into a visualization. The definition of this action includes three key query parameters: a data concept, a list of constraints and a summary function. *Data concept* defines what type of data entities a user is interested in. For example, in a real-estate domain, there may be multiple data concepts including houses, cities and schools. The *constraint list* defines the desired data entity properties. Finally, the *summary function* describes how the matching data should be summarized. For example, common summary functions include *Enumerate*, *Count* and *Average*.

**Table 1:** The set of actions identified in our survey of several different visual analysis environments.

Type	Intent	Parameters	Description
Annotate	Notes change	For each selected visual object: Constraint list Tag	A request to tag meta-information to the data represented by one or more visual objects.
Bookmark	Notes change	Visualization state Metadata (eg title, tags)	A request to save the current visualization for future review.
Brush	Visual change	Constraint list	A request to highlight a subset of visual objects.
Create	Notes change	Content	The creation of a new entry within a user's electronic notes.
Change-Metaphor	Visual change	Metaphor	A request to alter the active visualization technique.
Delete	History change	Action	Similar to undo, removes an action from the middle of a user's sequence of historically performed actions.
Edit	History change	Selected action Modified parameters	A request to modify the parameters of an action in a user's action history.
Filter	Data change Visual change	Constraint list	A request to reduce the data set being visualized.
Inspect	Data change Visual change	Constraint list	A request for 'details-on-demand' for a visual object.
Merge	Visual change	For each selected visual object: Constraint list	A request to combine the data represented by two or more visual objects into a single visual group.
Modify	Notes change	Selected note Content	A request to alter an existing observation in a user's electronic notes.
Pan	Visual change	Range constraint list	A request to scroll a visualization to a new location along an ordinal dimension.
Query	Data change Visual change	Data concept Summary function Constraint list	A request to bring additional data into a visualization.
Redo	History change	—	The inverse of undo, a request to re-perform an undone action.
Remove	Notes change	Selected note	A request to delete an observation from a user's electronic notes.
Restore	Data change Visual change	Bookmark	A request to restore a previously saved bookmark.
Revisit	History change	Selected Action	A request to return to an earlier stage of analysis.
Sort	Visual change	Dimension Order	A request to re-arrange the visual presentation along a particular dimension.
Split	Visual change	Split parameters For each selected visual object: Constraint list	A request to break apart the data represented by a set of visual objects into several new visual objects based upon some criteria, such as keywords or attribute values.
Undo	History change	—	Removes the most recently performed action from a user's sequence of historically performed actions.
Zoom	Visual change	Range constraint list	A request to scale a visualization to display a new range along an ordinal dimension.

As captured by the parameters of a *Query* action, the primary user intent is to change the set of data under analysis. This makes *Query* a data exploration action. In practice, however, a user's *Query* often results in changes to both the data set and visual presentation. For example, the real-estate analysis tool shown in Figure 3 automatically translates and scales a map to properly display the information returned by a user query.

Split: The *Split* action represents a user's request to re-cluster the data represented by a selected set of visual objects based on certain criteria, such as attribute values. The definition of *Split* has two key parameters. First, *split parameters* are used as the criteria for re-clustering. Second, for each selected visual object to be split, a *constraint list* is provided to define the scope of the action.

As the parameters of the *Split* action indicate, the primary user intent is to change the visual presentation without altering the underlying set of data itself. This makes *Split* a view exploration action. For example, the temporal analysis tool shown in Figure 4 shows a user's *split* action performed on calendar data. The single summary timeline is split into five smaller timelines, one for each person. However, while the user's

intent is simply to change his/her view, the timeline tool performs an additional background query to gather the personnel data needed to perform the required clustering.

Insight actions

Insight actions are performed by users as they discover or manipulate the insights obtained over the course of an analysis. A user's insights can be recorded, for example, as free-form notes or lists of visualization bookmarks. We classify all actions with the *notes change* intent as insight actions.

Based on the parameters of an insight action, we further classify them into two categories: visual insight actions and knowledge insight actions. *Visual insight actions* are those whose parameters explicitly reference objects within a visualization, such as *Annotate* and *Bookmark*. In these cases, users are explicitly marking visual objects (or entire visualizations) as related to their derived insights.

In contrast, *knowledge insight actions* relate to the expression and manipulation of new knowledge created by a user as the result of knowledge synthesis. This second category represents insight actions that have no direct reference to visual objects, such as *Create*. Knowledge insight actions are typically found in visual analytic systems that have a note taking or knowledge management component.^{19,27}

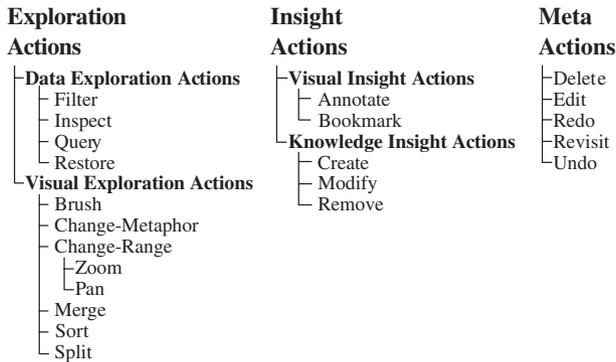


Figure 2: The action taxonomy contains three top-level categories: exploration actions, insight actions and meta actions.

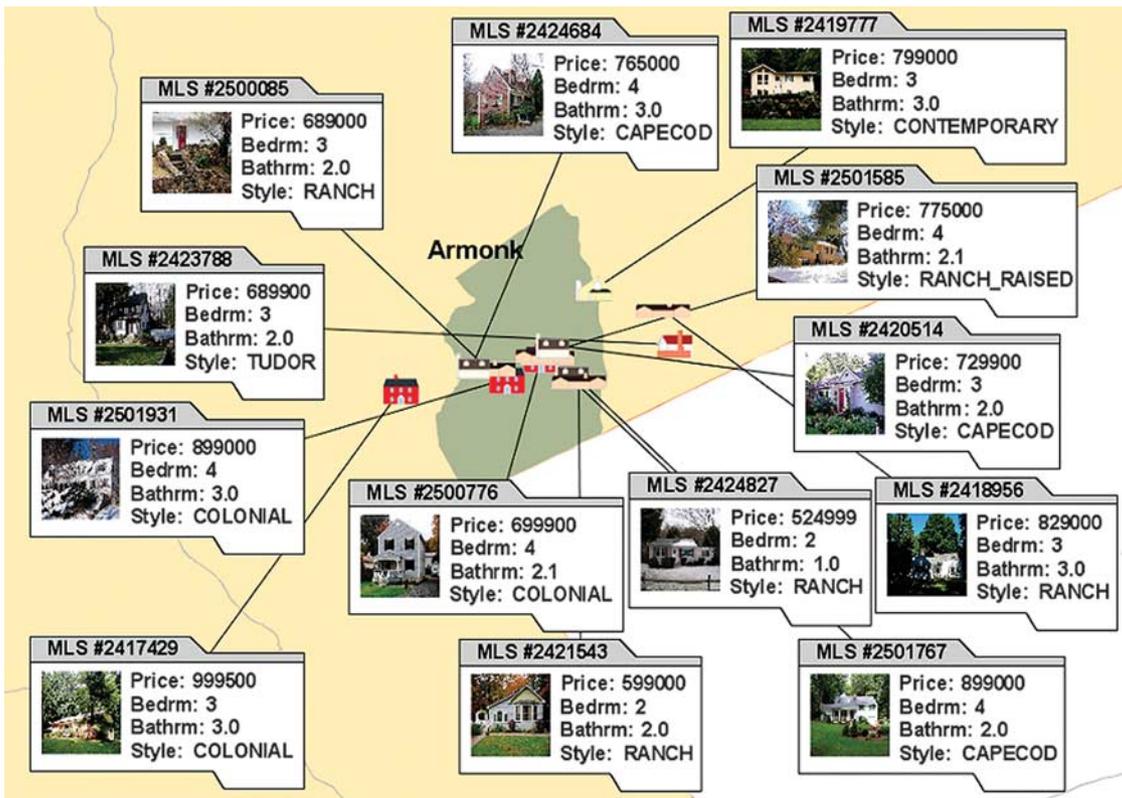


Figure 3: A *Query* action in this real estate analysis tool triggers a system response that changes both the data set under analysis as well as the visual presentation. However, because the primary intent of a *Query* is to change the data set, it is considered a data exploration action in our taxonomy.

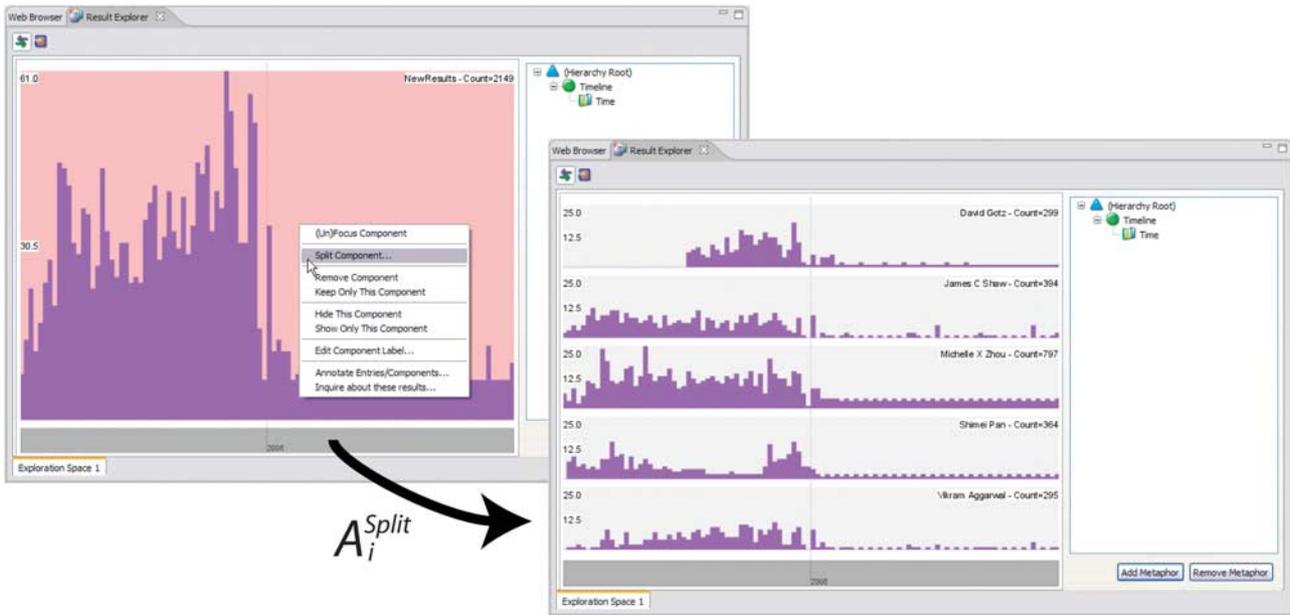


Figure 4: A *Split* action in this timeline-based analysis tool breaks apart a single timeline into several based on specific data attributes. The user's request is to change the visual organization of information already on display. For this reason, the *Split* action is a visual exploration action.

Meta actions

The third class in our taxonomy is *meta actions*. These actions, such as *Undo* and *Redo*, operate neither on the data set nor the visual presentation, but rather on the user's action history itself. In terms of intent, this class contains all actions with the *history change* intent.

Meta actions are important in part because they help distill the units of user activity that should constitute an action. For example, a user would not undo a single click, but may wish to undo a *Filter* or *Zoom*. In formulating our catalog of actions, we have attempted to define each at this 'undo' granularity.

An action-based sub-task representation

The actions identified in our catalog provide the semantic building blocks of insight provenance. However, it is insufficient for a visual analytic system to simply record a linear history of user performed actions. Such a history fails to capture how users combine multiple actions to accomplish higher-level sub-tasks. To provide more meaningful insight provenance, we must define a representation that captures how multiple actions combine to signal the accomplishment of a sub-task. In this section, we introduce the notion of *trails* as an action-based representation of sub-tasks.

Simple trails

A sub-task representation can be developed based on the observation (reported in the section Observations of User

Visual Analytic Activity) that user's perform sub-tasks in two distinct phases: an *exploration phase* followed by an *insight phase*. In terms of the action taxonomy, a user will accomplish a sub-task using a combination of exploration actions followed by an insight action. We refer to this basic pattern as a *simple trail*.

For example, assume that Bob from the sample scenario has just recorded a visualization bookmark in his notes to show that internet companies trade at a higher price-to-earnings ratio compared to other market sectors. To get to that point, Bob performed a series of actions: *Query* for financial stocks, *Split* by sector, *Sort* by price-to-earnings ratio and finally a *Bookmark* action to record the visualization snapshot in his notes. This sequence of actions documents the provenance behind Bob's conclusion.

Formally, we note a simple trail as τ^{simple} and define it using the regular expression in equation (2).

$$\tau = (A_i^{Exploration})A_i^{Insight} \tag{2}$$

Complex trails

In practice, trails are typically more complex than captured by the simple trail definition. First, analysts often perform meta actions to investigate alternative hypotheses. This causes branching in the logical structure of a user's activity. For example, after Bob identifies the two companies with highest price-to-earnings ratio, he might then analyze the first company in detail. Then, he could revisit the earlier stage of his analysis to begin performing a similar analysis for the second company.

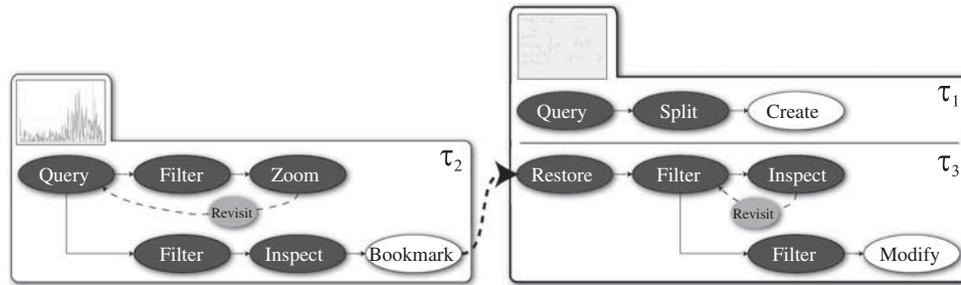


Figure 5: Trails can capture the nonlinear and progressive aspects of a user's analytic activity. When anchored to items in a user's notes, such as visualization bookmarks or fragments of text, they provide semantically meaningful insight provenance.

Moreover, analysts must often progressively chain together insights from multiple trails to satisfy specific sub-tasks. For example, Bob might retrieve an annotated set of high-growth companies to use as the starting point for a future sub-task. This creates a logical dependency between his future activity and his past actions that led to his discovery of the high-growth companies.

To capture the nonlinear and progressive nature of analytic activity, we expand our trail definition in two ways. First, we introduce meta actions to the exploration phase to allow nonlinear branching. Second, we introduce recursive references to trails within the definition to represent progressive insights. The recursive trail definition τ defined in equation (3) therefore captures the full web-like structure of a user's analytic activity.

$$\tau = (A_i^{Exploration} | A_i^{Meta} | \tau) A_i^{Insight} \quad (3)$$

This definition, in combination with our taxonomy, allows a visual analytic system to detect how sequences of user performed actions combine to satisfy individual sub-tasks. Systems can observe user behavior in terms of the actions in our taxonomy and automatically the web of trails that represent a user's analytic activity. Just as an insight action allows users to record *what* has been discovered, the automatically captured trails represent *how* a particular insight was generated. When paired with a user's notes, automatically extracted trails provide a comprehensive and logically meaningful model of insight provenance.

For example, consider the illustration in Figure 5. It shows three trails documenting two items in a user's notes that have been connected during the analysis process. First, the user created an entry in her notes as documented by trail τ_1 . Then, later in the analysis, the user bookmarked an interesting visualization discovered via trail τ_2 . Finally, after restoring the saved visualization and continuing same line of inquiry, the user modified her original notes after synthesizing a connection between her two insights. The nonlinear, progressive nature of this process is captured in the graph-like structure of the interconnected trails.

Practical Validation: HARVEST

Our characterization of analytic activity is motivated by the initial hypothesis that a semantics-based visual analytic system can automatically capture comprehensive and logically organized insight provenance without using application or domain heuristics. To validate this hypothesis, we have built a visual analytic system called HARVEST (see Figure 6) which uses semantic actions as a core design element for representing interaction activity. As a result, HARVEST is able to capture a semantic record of a user's analysis without any additional user work beyond their normal exploration and note taking behavior. Here we review how HARVEST captures and represents insight provenance and report some initial feedback to the system as gathered through interviews with both developers and analysts.

Insight provenance in HARVEST

HARVEST is a web-based visual analytic system that supports collaborative analysis. Individual users can upload and share data sets with co-workers. They can then visually explore the data to discover insights, which can then be shared with colleagues. As users interact with the system, HARVEST automatically captures a user's exploration behavior using our model of analytic activity and appends it as a trail to every visualization bookmark recorded by a user. When bookmarks are shared with collaborators, the saved trails are shared as well as a representation of insight provenance.

Our action-based model of user activity is a central element of the system's overall design. Every user facing component of HARVEST bundles low-level user events (for example, clicks and drags) into action-level expressions of user activity. They report all user behavior in terms of the actions identified in Table 1. These actions are monitored by an *Action Tracker* module which uses a rule-based approach to convert the linear sequence of user performed actions into a logical, graph-based representation of the user's analysis. While the algorithm itself is beyond the scope of this paper, the set of rules is

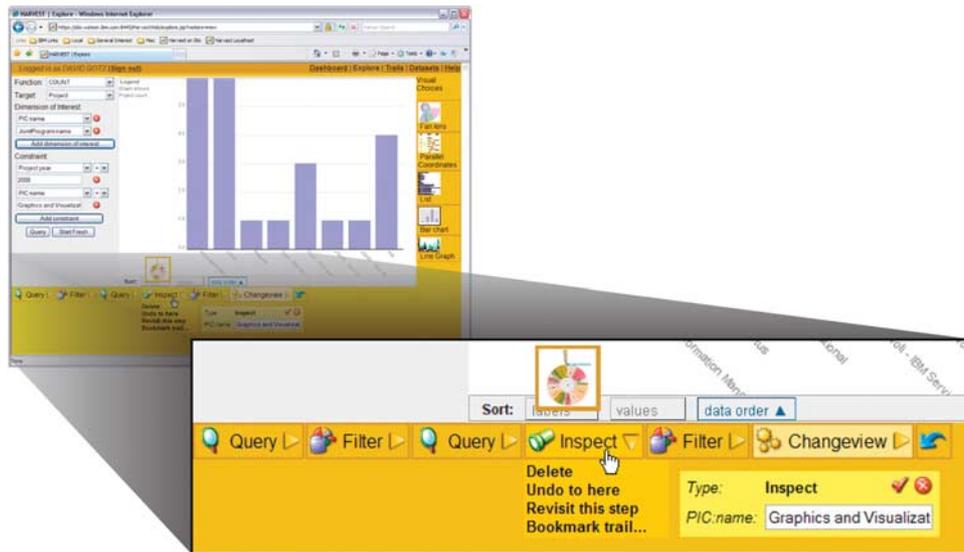


Figure 6: HARVEST is a web-based visual analysis platform that incorporates semantic actions as a core design element. As users interact with the system, a trail with the logical history of user performed actions is extracted and exposed in real time at the bottom of the screen.

derived based on the action taxonomy described in this paper.

As a user explores through his/her data, HARVEST displays a simplified segment of the action tracker’s trail representation as shown in Figure 6. The displayed trail allows users to perform meta actions such as undo, revisit and edit. In addition, when users bookmark a visualization to record an insightful view, the action trail is appended to the saved visualization state as insight provenance. When a bookmark is restored, the trail is restored as well providing context for the revisited discovery.

In addition, HARVEST allows users to look at the full trail behind a particular insight. For example, Figure 7(a) shows a visualization bookmark recorded by a user to show evidence of the bankruptcy of British Bank Northern Rock. It shows that more news articles about England mention Northern Rock than any other bank during the same time period. However, it is only through a review of the accompanying trail in Figure 7(b) that a collaborator can see that a spike in the volume of articles related to England in September 2007 led the user to focus in on the targeted subset of information. Moreover, the trail allows a user to easily revisit any of the intermediate steps along the way.

Developer feedback

HARVEST is a multi-metaphor visualization system which allows users to interactively switch between various views of data (for example, maps, timelines, scatter-plots, parallel coordinates, and so on). To integrate a new visualization widget into the HARVEST system, a developer

must ensure that all user interactions with the widget are modeled as actions and expressed through a defined API. We worked with several visualization developers and all of them found it straightforward and useful to model their components based on the semantic actions they provide to users.

For example, HARVEST includes parallel coordinate visualization tool that supports several actions. The developer of this visualization implemented a large variety of interaction mechanisms for triggering new commands, ranging from ‘drag and drop’ to popup context menus. However, using our action taxonomy, the developer was able to distill the systems functionality into four basic semantic behaviors: *Bookmark*, *Brush*, *Filter* and *Sort*.

Another contributor extended several widgets from the ManyEyes widget library³⁰ to allow additional interaction and customization capabilities. These extended widgets were then added to the HARVEST system. According this developer, it ‘was pretty straightforward’ to map the existing features of each tool to corresponding actions. ‘This was actually quite easy to do ... and it took a very little of my time.’ Moreover, the action catalog was sufficient for his widgets: ‘For the visualizations I was working on there were more than enough of the action types.’

Analyst feedback

We also observed several end users of the HARVEST system. The users analyzed enterprise data containing personnel, funding and strategy information describing ongoing projects within our company. In addition to observing their behavior, we performed interviews to

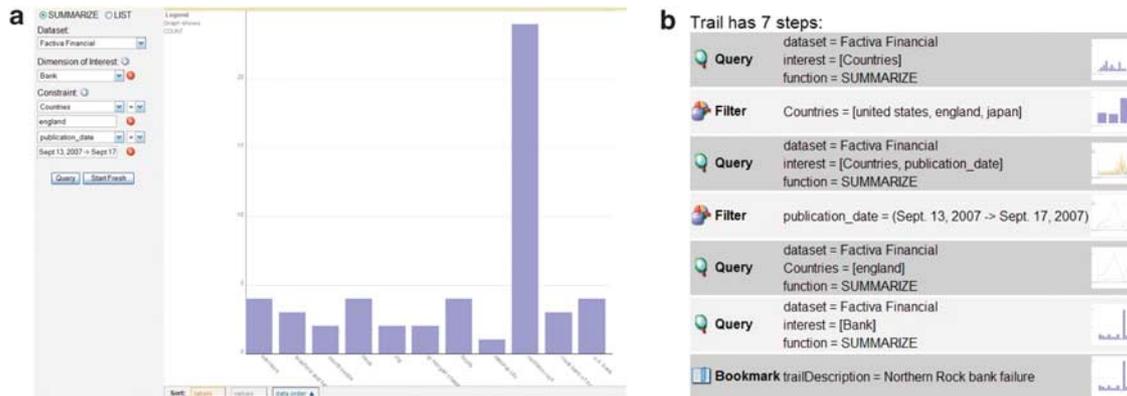


Figure 7: This HARVEST bookmark shows evidence of the British bank Northern Rock's descent into bankruptcy based on metadata extracted from a large set of 2007 news articles. (a) The visualization bookmark by itself provides no information about how the insight was found. (b) HARVEST automatically captures a trail along with the bookmark to document that the insight was found by exploring an unusual spike in articles related to England in September 2007.

gather feedback on HARVEST's action tracking and insight provenance capabilities.

The initial feedback from these users indicates that our approach aligns well with users' own mental models. For example, one user reported that actions 'match well with what I consider [my] individual steps.' This same user also mentioned that 'it is very helpful to have the actions updated and displayed in real time while I interact with the system,' stating that it helps give a 'clear picture of what state I am in.' In addition, users welcomed the use of action trails as provenance for recorded visualization bookmarks. When opening saved bookmarks, 'restoring the action trail [is helpful for] explaining how the visualization result was derived.'

Conclusion

Insight provenance is a critical requirement for many visual analytic applications. While previous work in this area has relied on either manually recorded provenance (for example, user notes and bookmarks) or automatically recorded event-based insight provenance (for example, clicks and key-presses), both techniques have fundamental limitations. Our aim is to develop a new approach based on a semantic model of user activity that combines the benefits of both approaches while avoiding their deficiencies. We therefore set out to identify a set of semantic units of user activity that could be used to capture insight provenance.

To identify an effective set of semantic building blocks, we have characterized a user's visual analytic behavior at multiple levels of granularity based on the semantic richness of user activity. Motivated by Activity Theory and our own empirical studies of analysts' behavior, we model user visual analytic activity at four levels: *tasks*, *sub-tasks*, *actions* and *events*.

Most critical in this model is the Action tier which we use to identify a set of generic and semantic units of user activity. We identified a set of common actions based on a review of several different visual analytic systems. We then characterized these actions along a number of dimensions, such as the type, intent and parameters of each user action. Based on this characterization, we developed a taxonomy that identifies three general classes of actions: *exploration actions*, *insight actions* and *meta actions*. The taxonomy can be used to both assist in inferring higher-level sub-tasks, and as a guide for defining new action types. To validate our approach, we incorporated actions as a core design element within our lab's visual analytic system, HARVEST. The initial feedback from both developers and analysts is promising.

Several challenges remain to address in future work. For example, more effective algorithms for inferring high-level logical structures from a user's performed actions are required. Another area we are exploring is new uses for insight provenance to assist in building more effective visual analysis environments. Finally, we must perform more comprehensive user studies to fully evaluate our approach.

References

- 1 Thomas, J.J. and Cook, K.A. (2005) *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE Press, New York.
- 2 Bavoil, L., Callahan, S. P., Crossno, P. J., Freire, J., Scheidegger, C. E., Silva, C. T. and Vo, H. T. (2005) Vistrails: Enabling interactive multiple-view visualizations. In: C.T. Silva, E. Groller, and H. Rushmeier (eds.) *Proceedings of IEEE Visualization*; IEEE Computer Society: Minneapolis, MN.
- 3 Eccles, R., Kapler, T., Harper, R. and Wright, W. (2007) Stories in geotime. In: W. Ribarsky, and J. Dill (eds.) *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology*; IEEE Computer Society: Sacramento, CA.
- 4 i2 Incorporated. *Analyst's Notebook*. www.i2inc.com.



- 5 Jankun-Kelly, T., Ma, K-L. and Gertz, M. (2002) A model for the visualization exploration process. In: Proceedings of IEEE Visualization; IEEE Computer Society: Boston, MA.
- 6 Kreuseler, M., Nocke, T. and Schumann, H. (2004) A history mechanism for visual data mining. In: Proceedings of IEEE Information Visualization; IEEE Computer Society: Austin, TX.
- 7 Lankenau, R. A., Eick, M. A., Decherd, A., Khalio, M., Paris, P. and Fugitt, J. (2006) Vast 2006 contest second place, corporate category: Decide. In: P.C. Wong, and D. Keim (eds.) Contest Results for IEEE Symposium on Visual Analytics Science and Technology; IEEE Computer Society: Baltimore, MD.
- 8 Nardi, B. A. (1996) *Context and Consciousness*. Cambridge, MA: The MIT Press.
- 9 Gotz, D. and Zhou, M.X. (2008a) An Empirical Study of User Interaction Behavior During Visual Analysis. IBM Research. Technical Report RC24525.
- 10 Gotz, D. and Zhou, M.X. (2008b) Characterizing users' visual analytic activity for insight provenance. In: D. Ebert and T. Ertl (eds.) Proceedings of IEEE Symposium on Visual Analytics Science and Technology; IEEE Computer Society: Columbus, OH.
- 11 Leont'ev, A. N. (1978) *Activity, Consciousness, Personality*. Englewood Cliffs, NJ: Prentice-Hall.
- 12 Gotz, D., Zhou, M. X. and Wen, Z. (2006) A study of information gathering and result processing in intelligence analysis. In: M.X. Zhou, and M.T. Maybury (eds.) Proceedings of Intelligent Users Interfaces Workshop on Intelligent User Interface for Intelligent Analysis; ACM: Sydney, Australia.
- 13 Patterson, E. S., Woods, D. D., Tinapple, D. and Roth, E. M. (2001) Using cognitive task analysis (CTA) to seed design concepts for intelligence analysts under data overload. In: Proceedings of the Human Factors and Ergonomics Society's 45th Annual Meeting; Minneapolis, MN.
- 14 Pirolli, P. and Card, S. (2005) The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In: Proceedings of the International Conference on Intelligence Analysis; McLean, VA.
- 15 Hampson, E. and Cowley, P. (2005) Instrumenting the intelligence analysis process. In: Proceedings of the International Conference on Intelligence Analysis; McLean, VA.
- 16 Scholtz, J., Morse, E. and Hewett, T. (2004) In depth observational studies of professional intelligence analysts. In: D.A. Vincenzi, M. Mouloua, and P.A. Hancock (eds.) Proceedings of the Human Performance, Situation Awareness, and Automation Conference; Routledge: USA.
- 17 Saraiya, P., North, C., Lam, V. and Duca, K. A. (2006) An insight-based longitudinal study of visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 12(6), 1511–1522.
- 18 Groth, D. P. and Streefkerk, K. (2006) Provenance and annotation for visual exploration systems. *IEEE Transactions on Visualization and Computer Graphics*, 12(6), 1500–1510.
- 19 Shrinivasan, Y. B. and van Wijk, J. J. (2008) Supporting the analytical reasoning process in information visualization. In: M. Czerwinski, A. Lund, and D. Tan (eds.) Proceedings of the SIGCHI Conference on Human Factors in Computing Systems; ACM: Florence, Italy.
- 20 Chi, E. H. (2000) A taxonomy of visualization techniques using the data state reference model. In: S. Roth, and D. Keim (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: Salt Lake City, Utah.
- 21 Chi, E.H. and Riedl, J. T. (1998) An operator interaction framework for visualization systems. In: J. Dill, and G. Wills (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: Research Triangle Park, NC.
- 22 Chuah, M. C. and Roth, S. F. (1996) On the semantics of interactive visualizations. In: S. Card, and S.G. Eick (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: San Francisco, CA.
- 23 Wehrend, S. and Lewis, C. (1990) A problem-oriented classification of visualization techniques. In: Proceedings of IEEE Visualization; IEEE Computer Society: San Francisco, CA.
- 24 Zhou, M. X. and Feiner, S. K. (1998) Visual task characterization for automated visual discourse synthesis. In: M.E. Atwood, C-M. Karat, A. Lund, J. Coutaz, and J. Karat (eds.) Proceedings of the SIGCHI Conference on Human Factors in Computing Systems; ACM Press/Addison-Wesley Publishing Co.: Los Angeles, CA.
- 25 Amar, R., Eagan, J. and Stasko, J. (2005) Low-level components of analytic activity in information visualization. In: J. Stasko, and M. Ward (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: Minneapolis, MN.
- 26 Amar, R. and Stasko, J. (2004) A knowledge task-based framework for design and evaluation of information visualizations. In: M. Ward, and T. Munzner (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: Austin, TX.
- 27 Gotz, D., Zhou, M. X. and Aggarwal, V. (2006) Interactive visual synthesis of analytic knowledge. In: P.C. Wong, and D. Keim (eds.) Proceedings of the IEEE Symposium on Visual Analytics Science and Technology; IEEE Computer Society: Baltimore, MD.
- 28 Shneiderman, B. (1996) The eyes have it: a task by data type taxonomy for information visualizations. In: M. Burnett, and W. Citrin (eds.) Proceedings of the IEEE Symposium on Visual Languages; IEEE Computer Society: Boulder, CO.
- 29 Wattenberg, M. (1998) Map of the market, <http://www.smartmoney.com/marketmap/>.
- 30 Viegas, F. B., Wattenberg, M., van Ham, F., Kriss, J. and McKeon, M. (2007) Many eyes: A site for visualization at internet scale. In: M. Chen, C. Hansen, C. North, A. Pang, and J. van Wijk (eds.) Proceedings of IEEE Information Visualization; IEEE Computer Society: Sacramento, MA.