A Knowledge-assisted Visual Analytics Framework for Knowledge Incorporation and Communication

Xiaoyu Wang*
Charlotte Visualization Center

Wenwen Dou†
Charlotte Visualization Center

Dong Hyun Jeong‡
Charlotte Visualization Center

Eric Bier§
Palo Alto Research Center

William Ribarsky¶
Charlotte Visualization Center

Remco Chang‖
Charlotte Visualization Center

ABSTRACT

Utilizing domain knowledge is crucial but less systematically understood in visual analytics systems. While many visual analytics systems have emphasized the use of domain knowledge, with few exceptions, the process of incorporating such knowledge is often specific to the domain, making it difficult to generalize to new problem areas.

In this paper, we propose a knowledge-assisted visual analytics framework to incorporate and support communication between both *domain* and *individual knowledge structure*. Our contributions are twofold: first, we review several knowledge management processes from the knowledge management community, and present these processes in the context that is relevant to knowledge-assisted visual analytics. second, we argue the importance of a knowledge mapping mechanism in supporting the knowledge communications. Based on the degree of users' involvement in creating such a mechanism, we further characterize three possible types of such knowledge mapping mechanisms.

1 INTRODUCTION

Even though our field has emphasized that the use of domain knowledge is crucial in facilitating users' analytical reasoning processes [**?**], with few exceptions, the process of incorporating such knowledge is often specific to the domain, making it difficult to generalize to new problem areas. Ideally, a system designer should be able to incorporate domain knowledge into visual analytic systems based on general design principles. Before we can effectively utilize the domain knowledge, we first need to understand what processes are necessary to incorporate such knowledge into a visual analytics system. While several high-level knowledge-assisted pipelines have been proposed to identify the role of domain knowledge in visual analytics systems [**?**], we still need a framework to provide general methods to computationally incorporate and utilize the domain knowledge.

Since identifying the knowledge incorporation methods is quite difficult, we begin with understanding the tangible artifacts of domain knowledge. In the field of knowledge engineering and management (KEM), researchers have designed various *knowledge structures* to computationally encapsulate and represent the domain knowledge [**?**] [**?**]. Specifically, for a human-computer system, Koenig [**?**] categorized the *knowledge structures* into two semantic subsets: the *domain knowledge structure* and the *individual knowledge structure*.

*e-mail: xwang25@uncc.edu
†e-mail: wdou1@uncc.edu
‡e-mail: dhjeong@uncc.edu
§e-mail: bier@parc.com
¶e-mail: ribarsky@uncc.edu
‖e-mail: rchang@uncc.edu

Further analysis in KEM indicates that the knowledge structures are computationally embodied in "formalisms". The formalisms exist in various knowledge management systems and are of different characteristics. Extending the categorization presented by Doconta [**?**] and Johson [**?**], we broadly classified the existing formalisms into three types: *ontological formalism, taxonomic formalism* and *task-related formalism*. While structurally distinct, these formalisms are often used together to represent both *domain* and *individual knowledge structure* [**?**].

Since several empirical studies [**?**] [**?**] [**?**] have demonstrated the benefits of utilizing both *domain* and *individual knowledge structure* to improve task performances, we argue that a framework which utilizes both types of knowledge structures could be sufficient to represent a computational usage of domain knowledge.

Therefore, in this paper, we propose a knowledge-assisted visual analytics framework to incorporate and support communication between both *domain* and *individual knowledge structure*. Our contributions are twofold: first, we review several knowledge management processes from the knowledge management community, and present these processes in the context that is relevant to knowledge-assisted visual analytics. second, we argue the importance of a knowledge mapping mechanism in supporting the knowledge communications. Based on the degree of users' involvement in creating such a mechanism, we further characterize three possible types of such knowledge mapping mechanisms.

Before further discussing the framework (Section **??**), we first discuss the definitions of basic concepts that we used to derive our framework. Specifically, we will try to address three intrinsic questions (see Section **??**): "What is knowledge structure?", "How is it computationally stored?" and "What is its current usage in visual analytics systems?". We will further detail the challenges in constructing such a knowledge-assisted visual analytics framework in Section **??**.

2 BACKGROUND AND RELATED WORK

2.1 Defining Knowledge Structure

While the incorporation of knowledge structures in visual analytics is still in its early stage, the research on utilizing knowledge structures has been well developed in the field of knowledge engineering and management. Much like the underlying motivation in visual analytics, research in knowledge management has focused on the key role of knowledge structure for decision-making in its own right. The researchers in that domain further attempted to deal with the management control regarding leveraging the knowledge structure at the intersection of human and computer [**?**]. While knowledge management covers a wide range of knowledge manipulations, our main interest lies in the core concepts of actions or processes to incorporate and support communication between different types of knowledge structures.

For the purpose of this paper, we extended the influential cognitive definition by Tulving [**?**] and defined the **knowledge structure** in the context of visual analytics as *a computationally organized structure of the relationship between task-related concepts in* *a particular problem domain*. Specifically, the ***domain knowledge structure*** represents knowledge that has been formally identified and generally agreed upon in a specific domain. In contrast, the ***individual knowledge structure*** is typically less formal and usually limited to the individual or a small group. It is what is explicitly stored or derived from the individuals explorations and analyses. Storage might be in a shoebox (evidence collection), via annotations, etc. Derivation might occur from logged user interactions or other means.

Towards computational approaches, the postulated central role of acquired knowledge has encouraged efforts to computationally externalize experts' knowledge into structures that can be utilized by computers and users [**?**]. Several techniques have been developed to help systematically construct such knowledge structures. Specifically, Koppen [**?**] developed an interactive questioning routine to extract the personal knowledge of domain experts. In addition, Langley et al. [**?**] demonstrated the possibilities of using computers to algorithmically compute such knowledge structures and facilitate human search for patterns in data repositories. More recently, Scardamalia et al. [**?**] presented practical approaches to capture and store computational artifacts into knowledge structures for later utilizations.

While these knowledge structures shared various structural formalisms, Koenig et al. [**?**] further categorized knowledge structures into *domain knowledge structure* and *individual knowledge structure*. Previous research has demonstrated the benefits of incorporating both types of knowledge structures in facilitating task performance. In both of their well-received empirical studies, Bierly et al. [**?**] and Zack [**?**], found that organizations, which acquire and share knowledge by combining individual- and domain- knowledge structures, tend to be more profitable and better performing. In addition, Choi et al. [**?**] further summarized such phenomena into a knowledge creation model that indicated the effectiveness of utilizing both types of knowledge structures in achieving higher analytical productivity.

Specific to the field of visual analytics, we have previously conducted research to understand how a visual analytics system could benefit from using knowledge structures. As reported in our previous work [**?**], through embedding the knowledge conversion processes [**?**] into visual analytics, we presented a process-based approach that enables visual analytics systems to incorporate knowledge structures. While we didn't specify details on how to incorporate and support communications between both types of knowledge structures, our previous work did demonstrate the theoretical and practical possibilities for incorporating knowledge structures into a visual analytic system.

2.2 The Formalisms of Knowledge Structure

In order to construct a framework to utilize both *individual and domain knowledge structures*, it is important to identify their structural formalisms and their corresponding computational artifacts. Extending the taxonomies presented by researchers in the knowledge management domain [**?**] [**?**], we summarized three typical formalisms of knowledge structures.

2.2.1 Ontological Formalism

By introducing links between concepts and their related information, this formalism tends to be more structurally complete for a focused area. A key idea of the *ontological formalism* is to carefully represent knowledge so that such a structure could help make inferences by traversing the various relationships between concepts. Therefore, this formalism relies on using conceptual links to form classes or hierarchies. These classes and hierarchies are further used to navigate quickly through the relevant region of knowledge structures, and to retrieve the required knowledge artifacts effectively [**?**]. Summarized in [**?**], this *ontological formalism* is often

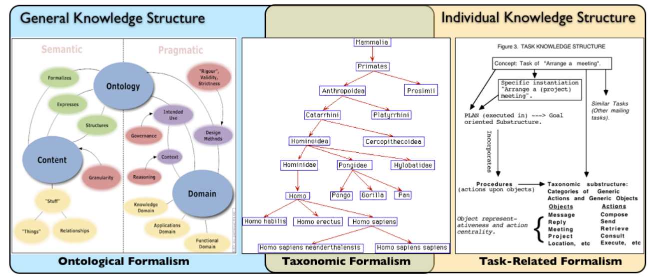

Figure 1: The three typical formalisms used to represent both *individual and domain knowledge structures*

used to rigorously represent *domain knowledge structures*.

The computational artifacts of this form of knowledge structure are often computed as formal and full-blown ontological knowledge models [**?**] [**?**](see Figure **??** (A)). Since construction of such knowledge structure is laborious [**?**], in practice, the *ontological formalism* could also built as a carefully connected collection of taxonomic structures. The wide use of this formalism can be seen in various applications from both fields of knowledge management [**?**] [**?**].

Several visual analytics systems have demonstrated the benefits of incorporating this formalism to access to knowledge embodied in *domain knowledge structures*. For instance, Dolan et al. [**?**] presented a visual analytics system that utilizes a formal biology ontology to facilitate access to knowledge about human disease genes. In addition, by interfacing with Visual Representation Ontologies, SemViz [**?**] presented its ability to generate appropriate visualizations automatically from a large collection of popular web pages for music charts without prior knowledge of these web pages. Our previous work [**?**] also demonstrated the general approaches to integrate *domain knowledge structure* into a visual analytics system; we further showed the utility of obtaining and presenting domain knowledge in visual analytic systems in the context of bridge management.

In summary, through the connection to ontological formalism, a knowledge-assisted visual analytics system can be enriched with the well-defined and organized knowledge, and provide domain specific inferences to users.

2.2.2 Taxonomic Formalism

A *taxonomic formalism* presents domain knowledge with minimal hierarchic or parent/child structure [**?**] and limited automated inferencing. Compared to *ontological formalism*, the *taxonomic formalism* focuses more on encompassing the structural relationship between concepts, but less on rigorously identifying the nature of the links between elements. In such formalism, the relationships between concepts are typically "subclass of" or "part of" relations. In general, the *taxonomic formalisms* are less laborious to construct than ontological formalism, and more easily extendable.

Examples of computational artifacts representing the taxonomic knowledge structure can be referred to as graph structure, informal ontology or relational databases (see Figure **??** (B)). In practice, the *taxonomic formalism* is widely adopted to represent both *domain knowledge structures* [**?**] [**?**] and *individual knowledge structures* [**?**]. Many visual analytics systems have shown the utility of using *taxonomic formalism* to access knowledge from both types of knowledge structures.

Focusing on retrieving taxonomic knowledge from a general domain, Bornor et al. [**?**] designed a taxonomy visualization and validation tool to support the semi-automatic validation and optimization of organizational schemas. In addition, through incorporating taxonomic structures and their synonymy relationships into visualization, Graham et al. [?] demonstrated the effectiveness of their visual analytic system in re-organizing large scale datasets and revealing domain specific taxonomic knowledge. Last but not least, through visualizing the taxonomic structure in the software visualization domain, Rhode et al. [?] presented their visualization system to guide user search and compare versions of software visualizations.

On the other hand, many visual analytics system have also demonstrated the utility to incorporate individual knowledge that is embodied in *taxonomic formalism*. Through the use of self-regulated learning taxonomy, Nussbaumer et al. [?] presented a set of visualization tools to effectively offer both guidance and grant control over the own learning process. More recently, we also presents an interactive visual analytics system to help users organize and retrieve business information, through obtaining knowledge from individuals' taxonomic document repositories [?].

In all, *taxonomic formalism* is widely used in visual analytics systems to provide accesses to knowledge from both *domain-* and *individual- knowledge structures*.

2.2.3 Task-related Formalism

Being a reflection of the structures found in reasoning tasks in the real world, a *task-related formalism* is a summary representation of different types of knowledge that are recruited and used in analytical task behavior [?]. While it is structurally less compact than the previous two types, such formalism more comprehensively covers knowledge concepts that are directly associated with task behaviors. When carried out together, these task behaviors present feasible solutions to address particular domain tasks [?]. Summarized in their theory about human-computer interaction, Card et al. [?] emphasized the importance of reasoning related knowledge structures in facilitating user's problem-solving processes.

As shown in Figure **??** (C), the computational artifacts of this kind of formalism can often be referred to as action rules [?], or human behaviors [?]. Due to its close representation of individual's tasks knowledge, the *task-related formalism* is often used to represent the *individual knowledge structures*.

The *task-related formalism* is used in visual analytics systems in two ways:

On one hand, many visual analytics systems have been designed to incorporate with this type of knowledge formalism. Specifically, with a declarative knowledge representation, Xiao et al. [?] presented a network traffic analysis system that supports the use of previous visual discoveries to enhance visual analysis. In addition, through communicating with a rule-based logic programming structure, Garg et al. [?] presented a visual analytic system that supports both model-driven data exploration, as well as data-driven model evolution. More recently, through the use of pre-defined scripting language, the Czsaw system [?] was shown to have utility in facilitating capturing and visualizing users' analysis processes.

On the other hand, previous research also has emphasized enriching *task-related formalism* based on considering users' individual task behaviors. In particular, Amar and Stasko [?] provided guidelines on how users' knowledge can be perceived through their interactions with visualizations, and proposed a high-level task taxonomy that could be used in analytical processes. In addition, Yi et al. [?] presented detailed categories measure and characterize insights gained through user interactions, and argued that interactions had become increasingly significant in the process of analytical reasoning.

Extending the above guidelines, many researchers have explored the possibilities of systematically capturing experts inputs during their reasoning process and extracting the task-knowledge embedded in those captured user inputs. Gotz et al. [?] characterized the users' analytical behaviors into four semantic tiers based on the semantic richness of the activities. Specifically focusing on analyzing the low-level interaction logs, Dou et al. [?] demonstrated the possibilities of recovering high-level semantic reasoning processes of domain experts. Their approaches provide significant results that experts' intents and strategies can indeed be recovered through examining their interaction logs.

In summary, this task-related formalism provides designers with a methodological approach to identifying and modeling reasoning knowledge. Great effort has been focused on not only utilizing this knowledge formalism, but also enriching it through the analysis of user's inputs.

3 Two Challenges to Address

The incorporation of the above formalisms in visual analytics system have show the great utility of both *domain* and *individual knowledge structures*. However, since many of this visual analytics systems are tailored to specific domains or tasks, it still remains an open question as to what general methods needed to incorporate and support communications within these types of knowledge structures.

Based on our extensive literature review in both visual analytics and knowledge management fields, we consider the challenges to answering this question as twofold:

One challenge is to come up with general knowledge process model to **incorporate** both *domain and individual knowledge structures*. Although many visual analytics systems have demonstrated approaches to incorporate either type of knowledge structure, with few exceptions, these methods are typically specific to particular task domains. Therefore, it can be quite challenging to generalize one method and extend it for other domains.

It is also a challenge to **communicate** between these two knowledge structures. Due to the possible mismatch of formalisms that are used in both types of knowledge structures [?], e.g. *ontological-* to *task-related formalism*, it is not always easy to achieve knowledge communication between these knowledge structures. Therefore, even though some systems could incorporate both types of structures, there may still be uncertainties about the communication between these knowledge structures.

To overcome these challenges, we propose a general knowledge framework to provide general guidelines to utilize and connect both *individual- and domain- knowledge structure* in visual analytics systems. As discussed in next section, we first review several knowledge management processes from the knowledge management community, and present these processes in the context that is relevant to knowledge-assisted visual analytics.

In our pursuit of a general knowledge framework for visual analytics systems, we have discovered several inspiring computational knowledge management models While many of these models focus specifically on certain organizations, the process model developed by Kucza [?] considers managing structured knowledge in a more general perspective.

4 A Knowledge Framework for Visual Analytics Systems

As shown in Figure **??**, Kucza detailed an abstract and generic knowledge management model to help incorporate and organize knowledge in a structured way, as well as to ensure that important structured knowledge is taken into account. In this model, five operational processes were identified to support an overall goal of incorporating knowledge in particular domain: identification of the need for knowledge, knowledge creation and validation, knowledge sharing, knowledge collection storage, and knowledge update.

Although each process represents a generic aspect of knowledge management, due to the scope of this paper, we only borrowed the three processes that were essential to knowledge incorporation and knowledge communication for visual analytics systems. We followed Kucza's definition and refer to any piece of knowledge as a

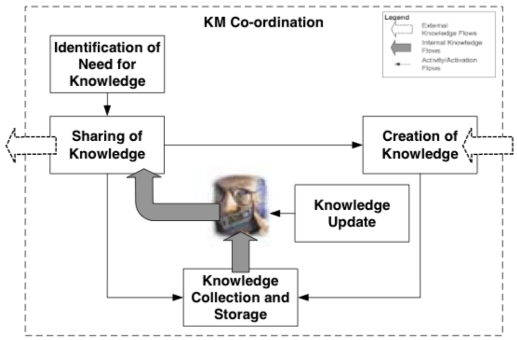

Figure 2: The original knowledge management modeled proposed by Kucza [?]. Note that both the knowledge creation and knowledge sharing processes are excluded from our visual analytics framework, due to the fact that they are specifically designed for the knowledge management domain and are not appropriate for the field of visual analytics.

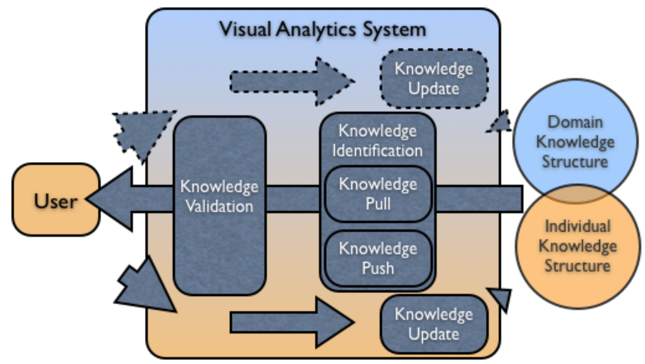

Figure 3: The three knowledge management processes that enable the exchange and combination of knowledge between the user, the visual analytics system, and the knowledge structures. Note that the knowledge updating process at the top is open, since the domain knowledge structure should not be casually updated by end-users.

"knowledge candidate" prior to its acceptance by users. As shown in Figure **??**, we further tailored these processes for the field of visual analytics, and characterized them as *knowledge identification, knowledge validation* and *knowledge update*. Together, these processes embody a knowledge flow between the user, the visual analytics system, and the knowledge structures.

Through the use of these knowledge management processes, our framework focuses on presenting guidelines to address both the knowledge incorporation and the knowledge communication challenges, and helps the design of a knowledge assisted visual analytics systems.

4.1 Addressing the Knowledge Incorporation Challenge

Grounded upon Kucza's analysis, we argue that the proper solution to knowledge incorporation challenges resides in realizing the knowledge flow embodied by the three knowledge management processes.

Based on user's interactions, a visual analytics system should be able to invoke the *knowledge identification* process to search and filter knowledge from both *individual knowledge structure* and *domain knowledge structure* (Section **??**). Once a knowledge candidate is identified, the visual analytics system should further utilize the *knowledge validation* process to validate that candidate for its accuracy and compatibility within both types of knowledge structures (Section **??**).

With a valid knowledge candidate, the system should then proceed to the *knowledge update* process (Section **??**). In our framework, this process takes effect in two places, the visual representations and the knowledge structures. On one hand, the visual analytics system should be able to validate the cost of visual changes caused by the knowledge update, and present user with proper notifications and minimize the interruption; On the other hand, the visual analytics system should be able to properly help users update the knowledge structures. While we argue that user shouldn't be allowed to casually update the *domain knowledge structure*, we think that it is important for visual analytics systems to update user's *individual knowledge structure* at runtime, and further maintain up-to-date knowledge mappings to balance communications between both types of knowledge structures.

While we describe these processes in a separate manner in next sections, it is important for a knowledge-asssisted visual analytics systems to utilize them together to establish a complete knowledge flow.

4.1.1 Process: Knowledge Identification

The goal of the *knowledge identification* process is to help user identify certain knowledge candidates that are suitable to their analytical processes. Depending on its different initiators, the design of the *knowledge identification* process can be practically considered as two sub-processes: the user's active *knowledge pull* and the visual analytics system's passive *knowledge push*.

Sub-process: Knowledge Pull With an interactive visual analytics system, the user should be able to actively pull knowledge from both *individual-* and *domain- knowledge structures*. This process is initiated by users through their active construction of visual queries at runtime. The output of this sub-process, knowledge candidates, can be used to invoke *knowledge update* process.

The main computations in this process are the search and filter for knowledge candidates in both types of knowledge structures. Depending on the type of knowledge structure that grands the knowledge pull, the design of this process should be considered differently:

Visual analytics system should allow users to pull knowledge by constructing visual queries that are directly tied to the *domain knowledge structure*. Since many existing visual analytics systems [**?**] [**?**] are tightly embedded with *domain knowledge structure*, the visual queries are typically automatically constructed through users' interactions with the system. For example, PatentVis [**?**] is a successful example for this design approach. By embedding patent-related knowledge with visual representations, this system allows users to interactively construct visual queries to explore and retrieve corresponding insights from a large repository.

On the other hand, the visual analytics system should also enable the user to explore knowledge stored their *individual knowledge structure*. Such a design approach can be seen in systems like Taste [**?**], where the visual analytics system relies on a pre-computed individual knowledge structure. By providing access to different facets of the stored documents, Taste enables users to interactively retrieve and organize business information based on their own knowledge about those artifacts.

Sub-process: Knowledge Push The *knowledge push* is an important variant of transferring knowledge to users known to need it [**?**]. In this sub-process, knowledge flow is triggered automatically without an explicit request from any knowledge seeker. Specific to visual analytic systems, the design of this process needs to focus on analyzing the knowledge towards a known need of users, either by finding repeatable operations or user-defined rules, and regularly transfer the knowledge to the identified users, with minimal interruptions. We have further identified three design considerations that are essential for this process:

Mapping: One crucial component to achieve the knowledge push process is the creation of proper mappings between collected *individual knowledge structure* and the *domain knowledge structure*. As presented by Gotz [?], users' prior inputs with a visual analytics system should be automatically captured and analyzed to create an individual profile. Such a profile can be used to reflect user's interests and their analytical goals [?], and should be further mapped to the *domain knowledge structure*. In practice, Zhang et al. [?] presented their knowledge mapping structure as an affinity matrix that helps to transform individual knowledge to domain knowledge, and further fulfill the knowledge push process. By collecting users' preferences through a reviewing/aligning process, the Knowledge Encapsulated Framework (KEF) [?] intends to customize its knowledge base for individuals' and present them with the most needed knowledge at runtime.

Timing: An important design consideration is the timing to trigger the knowledge push process. In their human cognition model [?], Green et al. argued that such timing should be determined by the changes of user's cognitive states. In practice, these kind of changes are normally related to tasks [?], and therefore could be monitored by the use of pre-defined task-related indicators, such as annotation or evidence creation [?].

Interruptibility: Following the research resented by Hudson et al. [?], we argue that the timing of knowledge push should also be constrained by the individual's interruptibility model. The degree of the interruption could be measured by the cost of updating visual representations (see **??**), or determined by pre-modeling the individual's interruptibility [?].

While there is no definitive way to implement knowledge push, we intend to use these three design considerations to gain an initial understanding that can be used to design such processes.

In summary, through the use of both knowledge pull and knowledge push, a visual analytics system should be able to help users identify knowledge candidates that may be needed to achieve their analytical goals. Due to uncertainty as to the validity of the knowledge candidate, any direct visual updates based on the *knowledge identification* process could lead to knowledge being lost or even knowledge conflict. Therefore, we argue that, prior to updating the visual representations, all the knowledge candidates should be validated through the knowledge validation process (see **??**)

4.1.2 Process: Knowledge Validation

The validation process is of great importance in the knowledge mapping structure. In this process, the knowledge candidates will be validated for their accuracy and compatibility. Visual analytics systems need to enforce this process in all of the knowledge processes at runtime. Following the well-received knowledge validation strategies presented by Bhatt [?], we further separate this process into three stages depending on the container of knowledge candidates at any given moment:

Knowledge Structure Stage: Once a piece of knowledge is identified as a candidate, the visual analytics system needs to assess its structural validity. This low level validation relies on the inquiries from visual analytics systems to the back-end knowledge structures to ensure the knowledge is valid and has been pulled or pushed correctly. Also, such validation needs to be carried out when the visual analytics system prepares to update the *individual knowledge structure*. Whereas the knowledge structure needs to be able to check if the knowledge update candidate is valid and conforms with previous rules or concepts. In the knowledge management community, structural level knowledge validation [?] [?] is considered an important research topic and connotes a sophistication that is beyond the scope of this paper.

Visual Analytics System Stage: When a knowledge candidate has passed structural level validation, visual analytics system will impose another level of validation on this candidate prior to updating the visual renderings. Following the well-rounded categorization of interaction and visual representation costs by Lam [?], this level of validation is directly associated with the user's cost of updating visual representations.

Specifically, at a low cost level (i.e. only part of the screen needs to be updated), the visual analytics system should be able to automatically validate the knowledge candidates either by algorithmically applying predefined criteria in the runtime or updating it. However, if the cost of change the visual representation is too high based on the pulled knowledge candidates, such as a drastic chang in graph layouts, user should be notified as to the possible changes prior to the visual actions. For example, such validation processes can be derived from Heer et al.'s [?] elaborated study on differentiate the cost of individual graph layouts.

User Stage: After passing the above two validations, the selected knowledge candidates should now be presented to the user through updated visual representations. However, at this level, the visual analytics system should be able to allow users to accept or dispute these knowledge candidates, based on their judgements on whether it is relevant to their on-going analytical reasoning processes. This validation presents an important feedback loop in the entire knowledge process, and would especially impact the construction of the user's individual knowledge structures. For example, the KEF system [?] presented user with the ability to review the suggested materials and accept/dispute based on user's preferences; through actively tacking the visual changes, the HARVEST system [?] also provides users capabilities to revert the visual updates.

In summary, by enforcing the above three stage of knowledge validation, we believe that a visual analytics system could present more accurate and suitable knowledge pieces for users to perform their analytical processes.

4.1.3 Process: Knowledge Update

The *knowledge update* process consists of updating visual representations and updating knowledge structures.

Once a valid knowledge candidate is identified, the knowledge update process should be triggered to update and change the corresponding visual representations. The aspect is integrated in most current visual analytic systems, such as [?]. Typically, when it is instructed by the users to perform knowledge discarding, joining/spitting or changing, the visual analytics system will automatically start the knowledge updating process. This process would then inform the visual interfaces to update based on the validated knowledge candidates.

The visual analytics system should also be able to properly update the utilized knowledge structures, and further maintain up-to-date knowledge mappings to balance communications between both types of knowledge structures. Since the construction of *domain knowledge structure* is more toward summarized facts of the domain knowledge, modifications performed on this type of knowledge structure without an experts supervision can endanger the validity of the entire knowledge structure. Therefore, in our framework, we only consider the updates that are needed for the *individual knowledge structure*.

Once a task knowledge is considered valid through the above validation process, the knowledge update process should first update the corresponding individual-knowledge structure, and then continue to compute a new mapping structure to keep the knowledge in both types of knowledge structures in sync.

To be able to keep the knowledge up-to-date and make valid knowledge mappings, the knowledge update process should be invoked when there are changes to either *individual-* or *domain-knowledge structure*. When the criteria are met, this process needs to update the knowledge mapping structures to balance those knowledge changes. Based on the structural similarity between the formalisms in both types of knowledge structure, the updating of the knowledge mapping structure can be categorized into three situations (see **??**).

In summary, although the above knowledge processes provide general guidelines to incorporate both *domain* and *individual* knowledge structure, supporting the communication between both types of knowledge structures still remains an important challenge to our framework. In the next section, we describe this challenge from a computational perspective and provide some guidelines to alleviating this challenge.

4.2 Addressing the Knowledge Communication Challenge

Communicating knowledge between *domain-* and *individual-* knowledge structures is a key activity for users' analytical reasoning processes. The research on understanding knowledge communication has long existed in both the fields of visual analytics and knowledge management. While there are no definitive guidelines to support such knowledge communication, several inspiring studies have demonstrated the possibility of addressing this challenge through the use of "knowledge mapping" [**?**] [**?**]. Although the term "knowledge mapping" presents a conceptual basis for knowledge communication, its computational artifacts are less tangible.

From a computational perspective, we argue that the mismatch of concepts stored in knowledge formalisms (see Section **??**) is the main cause for the difficulties in establishing knowledge communications. For a knowledge-assisted visual analytics system, this issue becomes even more significant due to its incorporation of various knowledge formalisms. The lack of proper knowledge communication may lead to knowledge being lost, or even knowledge conflicts, and could potentially hinder the user's analytical reasoning processes [**?**].

Therefore, to computationally support knowledge communication, we argue the importance of the creation of **knowledge mapping mechanisms** in knowledge-assisted visual analytics systems. Such mechanisms use computational artifacts, such as matrixes and graphs, to map concepts and support communications between both types of knowledge structures. Depending on the degree of user involvement in creating the mapping mechanism, there are three major designs (see Figure **??**):

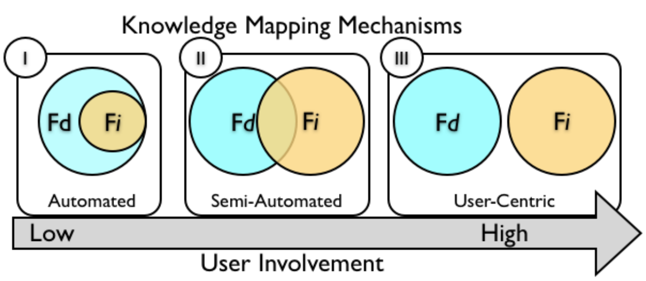

Figure 4: The three knowledge mapping mechanisms are used to support knowledge communication for mismatches of different formalisms that represents *domain-* (Fd) and *individual- (Fi)* knowledge structure . Depending on the degree of user involvement in creating the mapping mechanism, we consider them as automated, semi-automated and user-centric knowledge mapping mechanisms.

Fully Automated Knowledge Mapping: The existence of a fully-automated knowledge mapping mechanism requires the incorporation of identical formalisms in both *domain-* and *individual- knowledge structures*. To achieve this, both the concepts and structure of the formalisms used need to be identical. Typically, there should be an *inclusion relation* between these formalisms, as shown in Figure **??**(I).

When both *domain-* and *individual- knowledge structures* share identical formalism, their communication becomes automated and requires minimal or no user participation. The computation needed for this mechanism is the process to locate and match positions of the subset knowledge formalism within the superset. For instance, through connecting the individual biology ontology to the broader domain ontology, Dolan et al. [**?**] demonstrated the benefits of the automated knowledge mapping mechanism in facilitating the exploration and learning of human disease genes.

Semi-Automated Knowledge Mapping: The semi-automated knowledge mapping mechanism exists when the set of knowledge formalisms has an *intersection relation*, shown in Figure **??**(II). To create this semi-automatic mechanism, it is sufficient for the used formalisms to share concepts that can be related to each other, but are not exactly the same.

With users' explicit efforts to connect the concepts between the used formalisms, a knowledge-assisted visual analytic system should be able to create semi-automatic mathematical or computational model to achieve the knowledge mapping mechanisms.

Several visual analytics systems have previously demonstrated the possibility of creating semi-automated knowledge mapping mechanism with users' inputs. For instance, while using a taxonomic structure as its knowledge base, the KEF system [**?**] provides users with interfaces to map the relevance to individuals' knowledge, and stores those mappings as preferences for further knowledge notifications. In addition, by interactively involving the user in the process of ontology mapping, the CogZ system [**?**] presented a semi-automatic mechanism to support the user's decision making process and exploration of mappings.

User-Centric Knowledge Mapping: As shown in Figure **??**(III), the user-centric knowledge mapping mechanism is needed when the incorporated formalisms are fundamentally disjointed, or only single formalism is utilized. Therefore, in this situation, the only way to create a knowledge mapping mechanism is to rely on the users' explicit efforts on identifying their interests and preferences.

Typical user-driven techniques, such as guided explorations, can be applied to help establish such a knowledge mapping mechanism. For example, based on recordings of domain experts' exploration steps, the SYF system [**?**] created individual knowledge structures for each expert, and further reapplied their past paths of exploration on new data.

In summary, the creation of knowledge mapping mechanisms is key to establishing communications between different knowledge structures. While there are no definitive guidelines for designing such mechanisms, we have characterized three possible types based on the degree of user involvement, and provided design considerations for each.

5 Discussions

While our framework presents general processes to access and manage knowledge that is embedded in both *domain* and *individual knowledge structures*, our framework assumes that the knowledge already exists in a known structure. However, knowledge can come in a variety of forms: structured, semi-structured or unstructured [**?**]. In particular, in analytical reasoning, Davenport et al. [**?**] point out that much of the important knowledge needed to achieve the analytical goal is unstructured knowledge that exists mostly in the experts' minds. Since this type of unstructured knowledge is less tangible and therefore difficult to capture computationally, managing and utilizing such knowledge remains a great challenge.

From a traditional knowledge management perspective, the transformation from unstructured knowledge to structured knowledge typically involves the participation of knowledge modelers [**?**]. A knowledge modeler focuses on learning unstructured domain knowledge through their interactions and observations of domain experts' analytical processes. However, depending on the size of the domain and number of the experts that the modeler needs to consider, the resulting knowledge structure could sometimes be limited and incomplete. Therefore, in order to effectively process and manage such domain knowledge, new methods and tools are necessary to assist the collection and organizing of such knowledge into structures that can be easily managed and utilized.

With the advance of visual representations and interaction techniques, we propose that visual analytics systems could be an effective means to transform the users' unstructured knowledge into structured knowledge. Several visual analytics systems have already started to demonstrate capabilities in capturing and recovering domain knowledge. For example, by asking human coders to analyze experts' interactions logs, Dou et al. [?] demonstrated the possibility of recovering high-level semantic reasoning processes of domain experts. In addition, by encoding user's behaviors in a visual analytics system, Gotz et al. [?] presented a visual analytics system that can record and reapply users analytical processes. Last but not least, through the use of pre-defined scripting language, the Czsaw system [?] have shown capabilities in capturing and visualizing the users' analysis processes. In summary, while using visual analytics system to transform unstructured knowledge into structured knowledge is still at its early stage, it has already shown great potential.

Therefore, we propose that a potentially fruitful solution for transforming unstructured knowledge is likely an integration of interactive techniques with automated modeling. The challenge of such integration lies in identifying the roles and relationships between the domain concepts and interactions. At this point we do not have a clear outline on how to accomplish such integration that could leverage the strengths of both methods, but this is an important challenge that we will look to address in the near future.

6 Conclusion and Future Work

Although many visual analytics system have demonstrated the great benefits of incorporating domain knowledge in facilitating users' analytical processes, with few exceptions, the process of incorporating such knowledge is often specific to the domain, making it difficult to generalize to new problem areas.

In this paper, we propose a knowledge-assisted visual analytics framework to incorporate and support communication between both *domain* and *individual knowledge structure*. Extending on Kucza's knowledge management process model [?], our framework provides three knowledge management processes that enable the exchange and combination of knowledge between the user, the visual analytics system, and the knowledge structures.

Our contributions for this paper are twofold: first, we review several knowledge management processes from the knowledge management community, and present these processes in the context that is relevant to knowledge-assisted visual analytics. second, we argue the importance of a knowledge mapping mechanism in supporting the knowledge communications. Based on the degree of users' involvement in creating such a mechanism, we further characterize three possible types of such knowledge mapping mechanisms.