

IdeaCrepe : 履歴レイヤー構造を持つ知識創造支援ツール

IdeaCrepe: Creativity Support Tool With History Layer

中園 長新 †
Nagayoshi NAKAZONO †

三末 和男 ‡
Kazuo MISUE ‡

田中 二郎 ‡
Jiro TANAKA ‡

† 筑波大学 第三学群 情報学類

‡ 筑波大学大学院 システム情報工学研究科 コンピュータサイエンス専攻

†College of Information Sciences, ‡Graduate School of Systems and Information Engineering,
University of Tsukuba.

E-mail: †zono@iplab.cs.tsukuba.ac.jp, ‡{misue, jiro}@cs.tsukuba.ac.jp

Abstract

We developed a creativity support tool called “IdeaCrepe” which manages history of ideas by using layered structure. Handling ideas with a creativity support tool able to record the history of ideas, are very useful to help understand these ideas, their evolution through time and their inter-relations. We think that looking back over the history of creativity contributes to the further development of ideas. In this research, we introduce the concept of “layered structure” in order to manage the history. By using layered structure which is based on the concept of time series, we can look back over the past states of ideas, and check their development. These feature are allowed by managing events such as making of new ideas, modifying of them.

1 はじめに

我々は履歴をレイヤー構造で管理する知識創造支援ツール “IdeaCrepe”を開発した。このツールは、過去のアイデアの状態が簡単に把握できる履歴管理機能を持っている。

我々は普通の生活において様々なことを思いつき、そして忘れていく。しかしながら一方では、思いついたアイデアの断片を忘れる前に記録したり、その断片を元に発想をふくらませていきたいという欲求がある。そのため、たくさんのアイデア断片を産み出し記録してそれらを発展させていく知識創造活動の技法が多く考案され、また実際に活用されてきた [15]。

しかしながら知識創造活動は断続的に実施されることも多く、何回も活動を積み重ねていくと産み出されたデータがいつのものであったか、あるいは変更される前のデータがどのようなものであったかを思い出すことが難しくなる。過去の状態を参照しようとしても、その状態は上書き保存によって失われていることが多く、参照できない場合が多い。

そのため本研究では、このような履歴の振り返りができる知識創造支援ツールの開発を行った。履歴の振り返りが我々の知識創造にどのような効果をもたらすのかについては現在明らかにされていないが、直感的な考え方としては、過去の傾向や失敗あ

るいは成功を振り返ることで今後の活動がよりよくなると考えられる。そのような振り返りを支援する機能を知識創造支援ツールに導入し、試用と考察を行った。

2 知識創造支援ツールにおける履歴管理

2.1 知識創造活動とその手法

知識創造を支援するための技法としては、今日に至るまで数多くの提案がなされてきた。有名なものとしては川喜田二郎の提唱した KJ 法¹[14]、Alex Osborn が創始したブレインストーミング、Tony Buzan の発案による Mind Maps²の技法などがあり、広く用いられている。実際に紙とペンを用いて行う知識創造活動の様子を図 1 に示す。この例では、アイデアの断片は付箋紙に記入されて大きな紙の上に貼られており、付箋紙はペンで描かれた線によって関係づけられている。これらは広い意味でネットワーク図と見なすことができる。このようなネットワーク図は KJ 法 (カードを用いる) や Mind Maps(枝の上にアイデアなどを描く) をはじめとする多くの知識創造技法の基礎に通じるものである

¹KJ 法は株式会社川喜田研究所の登録商標である。

²Mind Maps は Buzan Organisation, Ltd. の米国および英国における登録商標であり、日本では「マインドマップ」と表記される。

ため、本研究ではネットワーク図を用いた知識創造支援ツールに焦点を合わせる。

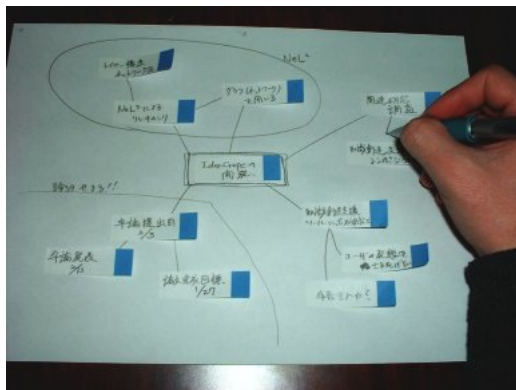


図 1: 紙とペンによる知識創造活動の様子

2.2 履歴の振り返りから新しいアイデアへ

知識創造支援で大切なことは数多く考えられるが、本研究ではその一つとして、過去に行った知識創造活動の振り返りに着目する。「温故知新」という言葉に代表されるように、我々は古来から自分たちの過去を振り返り、その振り返り活動を通して新しい活動における指針を見いだしてきた。これは計算機上で行われる、現代の知識創造活動にも十分に当てはまると考えられる。

多くの知識創造活動は、一つのテーマに対して何回も考察を重ねていくことで行われる。そのため、知識創造活動の成果は日々変化し、成熟に向かっていく。このように時系列に沿って発展していく活動において、アイデアがいつどのような経緯で生まれたのかといった過去の履歴を振り返ることは、今後の活動指針を決める上で重要な要素であると考えられる。過去にどのようなアイデアが生まれ、そのアイデアに対してどのような考察を試みたのか、そしてその結果がどうであったのか。そのような履歴を把握することで、過去に行った活動が新しい活動の原動力となる。

一般的な多くのアプリケーションで採用されている「上書き保存」の手法は、常に最新版のスナップショットを保存しているといえる。過去の状態を参照したい場合は各時点のデータ（スナップショット）を上書きせずに保存し、各データを比較することが必要となる。しかしながらこのような機能はあまり実装されていない。また、仮に各時点のスナップショットを保存したとしても、それらを比較して変化（差分）を発見することは容易ではなく、ユーザは複数のスナップショットを同時に閲覧する必要がある。

この問題は知識創造支援ツールにも当てはまる。そこで本研究では、履歴を管理できるデータ処理の枠組みを実装した知識創造支援ツール“IdeaCrepe”を開発した。

3 履歴レイヤー構造

本研究では、知識創造活動の履歴管理手法としてレイヤー構造 [9] を導入する。これは知識創造活動によって生成されたネットワーク上のデータを一枚の図としてではなく、更新時期の違いによってレイヤー別に管理し、それらの表示や色分けを柔軟に行うことでネットワーク図から多くの情報を読み取れることを可能にするものである。

3.1 レイヤー構造によるネットワークの履歴管理

本研究では、知識創造活動によって産み出されたアイデアをネットワーク図で表現する。我々はこのネットワーク図に対して図 2 に示すようなレイヤーの概念を導入した。これは OHP シートのような透明なシートを積み重ねたものとして考えることができる。ある時点または期間における更新は一枚のシート（レイヤー）に記録され、その次に行われた更新は既存のレイヤーの上に新しいレイヤーを重ねてそれに記録する。それぞれのレイヤーには各時点・期間での更新箇所すなわち差分のみが記録されていくことになり、これらを用いて次のような閲覧操作が可能となる。

- ある時点までのレイヤーを全て積み重ねることで、その時点におけるネットワーク図を見ることができる。レイヤーを順に積み重ねることで過去から未来への変化をたどることができ、逆にレイヤーを順に取り除くことで未来から過去への変化をたどることができる。
- 任意のレイヤーを単独で閲覧することで、その時点におけるネットワークの変化を見ることができる。
- 連続するレイヤーを複数枚重ね合わせることで、ある期間におけるネットワークの変化を見ることができる。

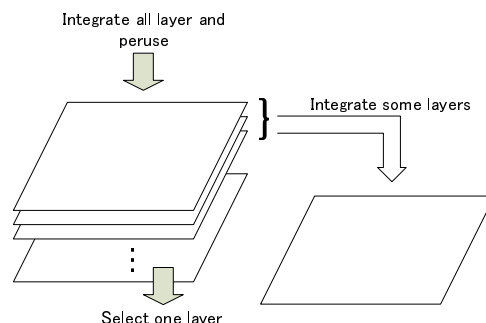


図 2: レイヤー構造の概念図

3.2 知識創造支援ツールへの応用

レイヤー構造は従来、時間軸をある一定の期間(たとえば1日、1年)ごとに区切り、各区間にレイヤーを割り当てるものであった。しかしながら多くの知識創造活動は断続的に行われる活動であり、一定期間ごとに区切るのには難しい。また、一つの期間内で同一オブジェクトに複数のイベント(ノードやエッジの追加・削除など)が発生した場合、この形式では全てのイベントを保存することができないという問題が生じる。

そこで本ツールでは、レイヤーは一つのイベントを一枚とし、イベントが発生するたびにレイヤーが追加されていくという概念を採用した。一枚のレイヤーにはただ一つのイベントが保存される。そのレイヤーを積み重ねていく活動は知識創造活動に等しい。この方法により、イベントがどのような順序で発生したかが明確になり、より確実に履歴管理を行うことができるようになる。

このように一つ一つのイベントをそれぞれ1枚ずつのレイヤーに記録し、履歴を管理する手法を履歴レイヤー構造と呼ぶこととする。

4 知識創造支援ツール“IdeaCrepe”

履歴レイヤー構造を備えた知識創造支援ツールとして、我々は IdeaCrepe³を開発した。

4.1 アイデアを扱う形式

多くの知識創造支援ツールと同様に、IdeaCrepe はネットワーク図を用いてアイデアを管理する。グラフによってアイデアを扱う場合、紙とペンによる知識創造技法と計算機上の作業の間には、一般的に次のような対応付けがなされる。

- 一つのアイデアはカードや付箋紙など、移動操作を行える紙の断片に記録することが多い。計算機上のネットワーク図では一つのノードとなり、画面上に描画される。
- アイデア同士の関係は、カード同士を線で結ぶことで表現される。計算機上でも同様に、ノード間にエッジを描画することで表現する。

これらの対応付けは一般的かつ直感的であるため、IdeaCrepe においても同様の形式を採用した。

³この名称には、アイデアがクレープ生地のように広がっていくこと、そしてミルクレープのようにアイデアが積み重なって大きな成果を生み出すことをサポートするツールでありたいという願いが込められている。

4.2 履歴レイヤー構造

知識創造支援ツール上で、ユーザは様々な操作を行う。その操作例としては次のようなものが考えられる。

- アイデア断片をノードとして生成する(ノード生成)
- アイデア断片間の関係線としてエッジを生成する(エッジ生成)
- ノードやエッジのラベルを変更する(ラベル変更)
- ノードの位置を移動する(ノード移動)
- ノードやエッジを削除する(削除)

本研究ではこれらの操作を「イベント」と呼ぶ。なお、ノードとエッジを総称して「オブジェクト」と表現する。

従来の知識創造支援ツールの多くは、アイデアを保存する際は既存データに上書きして保存するという手法をとっていた。この手法では、たとえばあるノードのラベルを変更したり位置を移動するといったイベントが発生した場合、過去のラベルや位置を参照することは不可能になる。

本ツールではデータの保存形式として履歴レイヤー構造を採用する。

4.3 アイデア記録のデータ形式

IdeaCrepe 上のイベントは XML 文書として記述する。レイヤー構造ネットワークは本来、レイヤー構造を持つために layer 要素を定義するが、IdeaCrepe は一つのイベントが一枚のレイヤーに対応するため、レイヤーに対応する要素を明示的に用いる必要がない。このような XML 文書の例を図3に示す。

まず、それぞれの親要素について説明する。node および edge 要素は、それぞれノードとエッジを定義する。その他の親要素(move, del, relabel)はイベントを定義しており、それぞれ移動、削除、ラベル変更を定義する。

次に、それぞれの親要素に共通する子要素・属性を説明する。date 属性はイベントの発生時間を記録し、id 属性はイベントの ID を記録する。この ID はイベントごとに独立してカウントされる。label 要素は各オブジェクトのラベル文字列を格納する。position 要素はオブジェクトの位置を指定するものであるが、node 要素や move 要素に対しては x, y 座標を持ち、edge 要素に対しては始点と終点ノードの ID を格納するようにした。これは、ノードは座標を指定することで位置が絶対的に定められるが、エッジは両端のノード位置に依存して位置が変化する可能性があるためである。

```

<?xml version="1.0" encoding="UTF-8"?>
<graph>
  <node date="1138036083031" id="0">
    <label>IdeaCrepe</label>
    <position x="311" y="235"/>
  </node>
  .....
  <edge date="1138036148453" id="0">
    <label>nextstep</label>
    <position from="7" to="10"/>
  </edge>
  .....
  <move date="1138036148502" id="0">
    <position x="248" y="420"/>
    <ref ref_com="node" ref_id="5"/>
  </move>
  .....
  <del date="1138045728102" id="0">
    <ref ref_com="edge" ref_id="14"/>
  </del>
  .....
  <relabel date="1138045879910" id="0">
    <label>NewLabel</label>
    <ref ref_com="node" ref_id="8"/>
  </relabel>
</graph>

```

図 3: XML による IdeaCrepe イベントの記述例

イベントを定義する親要素は、子要素として ref を持つ。これはイベントを適用するオブジェクトを指定する要素で、ref_com 属性で適用先のオブジェクト種類、ref_id 属性で適用先の ID を指定する。この組み合わせにより、ID がイベントごとに独立してカウントされていてもオブジェクトを一意に識別できる。

4.4 レイヤー構造を用いたアイデアの記録

本ツールが持つ最大の特長は、オブジェクトに対して移動、削除、ラベル変更といった操作を行っても、過去の情報が保存され参照できるという点にある。そのため、XML 文書において一度作成された要素（イベント）は変更されたり削除されたりすることはなく、操作による変更は別の要素を追加することによって記録される。

例えば、ラベル「January」を持つあるノードを生成した後、ラベルを「February」に変更するという操作を考える。最初にノードを生成すると図 4 のような node 要素が XML 文書に追加される。ここでラベル変更を行った場合、上書き保存を行うと図 5 のように変更され、以前のラベルが「January」であったという情報は失われてしまう。IdeaCrepe はこのような保存方法ではなく、新規に relabel

要素を追加する。すなわち、node 要素が保存されている以前のレイヤーに対して、relabel 要素を持つ新規レイヤーを重ねることで情報の更新を行う（図 6）。

```

<node date="1138036083050" id="1">
  <label>January</label>
  <position x="271" y="175"/>
</node>

```

図 4: ノード生成により追加される XML 要素

```

<node date="1138036084175" id="1">
  <label>February</label>
  <position x="271" y="175"/>
</node>

```

図 5: 上書き保存による XML 要素の変更

```

<node date="1138036083050" id="1">
  <label>January</label>
  <position x="271" y="175"/>
</node>
.....
<relabel date="1138036084175" id="3">
  <label>February</label>
  <ref ref_com="node" ref_id="1"/>
</relabel>

```

図 6: レイヤー構造を用いた XML 要素の追加

4.5 ツールのユーザインタフェース

開発したツール「IdeaCrepe」のスクリーンショットを図 7 に示す。本ツールは基本的な操作を全てメインウィンドウで行う。ウィンドウ右の部分が描画キャンバスであり、知識創造活動はこの領域にオブジェクトを配置することで行う。ウィンドウ左の部分にはファイル操作や表示変更に対応するボタンを配置し、基本的な機能はクリックするだけで呼び出せるようにした。これはメニュー操作などの煩雑な操作をできるだけ減らすことで、知識創造中のユーザの思考をできるだけ妨げないようにする配慮である。

4.6 ユーザの操作

本ツールは知識創造活動を支援するものであり、活動を妨げるものであってはならない。そのため、

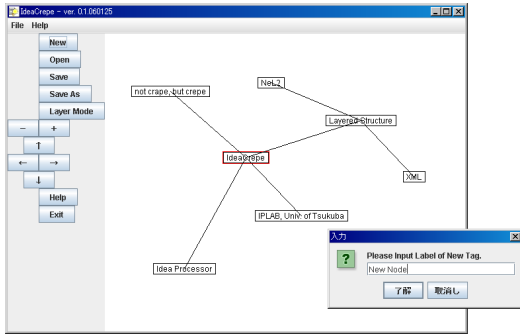


図 7: IdeaCreppe のスクリーンショット

オブジェクト生成や選択などといったモード切替を意識せずに操作できるように配慮した。本研究ではユーザの知識創造活動を妨げないためにユーザインタフェースが備えるべき要件として、次の2点を考慮した。

- 操作はユーザにとってできるだけ簡単なものであり、作業を選択するための労力ができるだけ少ないものであること
- 操作はユーザにとって直感的であり、深く考えることなく自然に発想できる動きで実行できるものであること

前者の例としては、メニュー階層を深く辿るよりもボタンをクリックの方が操作としては単純である。後者の例としては、キャンバスをクリックすればそのクリックした位置にノードが生成されるような事例が挙げられる。

なお、操作に用いるデバイスとしてはマウスとキーボードの併用を想定する。

実際に IdeaCreppe を利用するユーザは、次のような操作を組み合わせることで知識創造活動を行っていく。

ノード生成

アイデアの断片を思いついたユーザはキャンバスの任意の位置をクリックする。するとダイアログボックスが現れ、ここにラベルを入力することでアイデア断片を記録した新しいノードが生成される。このノードはキャンバスのクリックした位置に生成される。

ノード移動とエッジ生成

紙とペンによる知識創造活動において、ノードに対応するカードを移動させる場合、我々はカードを指でつかみ、引きずることで配置を変更する。カード間に関係線を生成する場合は、カード間でペンを引きずることで線を描画する。このように、ノード移動とエッジ生成は「引きずる」という似通った操作で行っており、この操作は計算機上でのマウスドラッグと対応する。移動とエッジ生成の違いは、

手にペンを持っているかどうか、そして引きずった先に別のカードがあるかどうかである。この作業を計算機上で実現する場合、ペンの有無はモード変更をユーザに意識させることになり、操作が複雑になる。そこで本ツールでは、引きずった(ドラッグした)先に別のノードがあるかどうかで操作を判別することとした。

キャンバス上に配置されたノードをキャンバス上でドラッグすることにより、ノードの位置を変更することができる。このとき、移動したノードにエッジが接続している場合、それらの配置も自動的に変更される(図8)。ノードをドラッグした先がキャンバス上ではなく他のノードだった場合、2つのノード間にエッジが生成される(図9)。

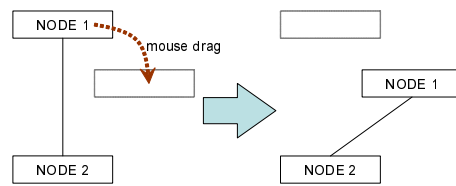


図 8: キャンバスへのマウスドラッグによるノードの移動

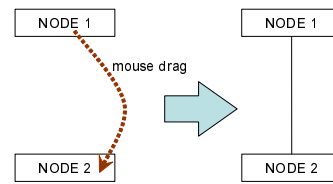


図 9: ノード間のマウスドラッグによるエッジの生成

ラベル変更とオブジェクト削除

ラベルの編集については、編集するオブジェクトを指定することによって行われる。本ツールでは移動時などの選択と区別するため、ノードやエッジをダブルクリックすることでラベルが編集できるようにした。

また、削除に関しても削除するオブジェクトを指定することで行われるが、描画とは異なる操作であることだけはユーザが意識する必要があると考えられる。そこで、左クリックではなく右クリックすることで削除を行うことができるようにした。

このように、オブジェクトの生成、編集に関してはユーザはツールのモードを意識することなく、簡単なマウス操作の使い分けで操作を行うことができる。

履歴レイヤー表示モードへの移行

ウインドウ左の“Layer Mode” ボタンをクリックすることで、IdeaCrepe は履歴レイヤー表示モードに移行する (図 10)。このモードはユーザの期待する操作が通常モードと明らかに違うため、ボタンをクリックして明示的にモード変更が必要であるように実装した。このモードはスライダ操作によってレイヤーの表示状態を変化させる。

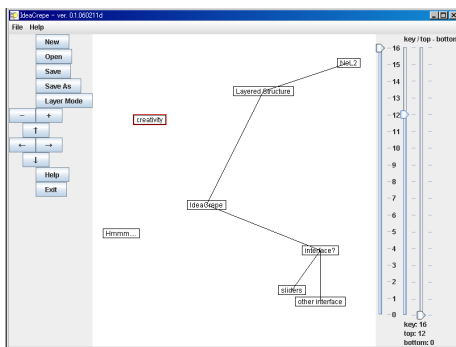


図 10: 履歴レイヤー表示モード

4.7 レイヤー構造を用いたアイデアの可視化

レイヤー構造を用いてアイデアを XML 文書に保存することにより、以前のデータを保持したままにアイデアを更新することが可能になった。しかしながらこの手法は、たとえば 4.4 の例では node 要素 (ノードのデータ) と relabel 要素 (イベントのデータ) のように、一つのオブジェクトに対して複数の要素が関連することが想定される。描画時には最新あるいは指定した期間の情報が表示される必要があるため、ノードやエッジのデータをイベントデータで上書きして更新しなければならない。また、IdeaCrepe は一つ一つのイベントをそれぞれ別々のレイヤーとして保存するため、イベント発生時に画面描画とデータ更新を同時に行う必要がある。また、既存のイベントを変更する代わりに変更を表す新たなイベントを追加するため、オブジェクトは複数のイベントに上書きされる可能性がある。

このような仕様を満たすため、IdeaCrepe では図 11 のような流れでデータを扱う。さらに、イベントの更新時刻を XML の date 属性に格納することで、時系列に沿ったイベントの管理ができるようにした。本研究では、1970 年 1 月 1 日 0 時 0 分 0 秒 GMT を起点とした時間をミリ秒で表した値を用いている。なお、この値を例えば「2006-01-10 15:43:24 GMT」などの文字列とし、この文字列を扱えるようにプログラムを改変すれば、より可読性の高い XML 文書とすることも可能である。

XML 文書を読み込むと、IdeaCrepe はそのデータを IdeaEvent クラスのインスタンスに記録し、ハッシュテーブルに格納する。格納されたデータのうち、

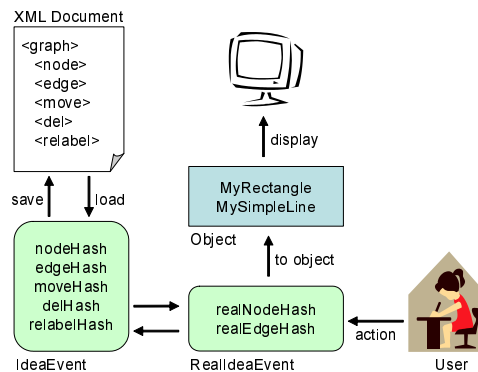


図 11: IdeaCrepe でのイベントの流れ

移動、削除、ラベル変更のイベントに関しては対応するノードやエッジの情報に上書きしてデータを更新し、新しいハッシュテーブル (RealIdeaEvent) に格納する。このハッシュテーブルに格納されたデータを Java の Graphics オブジェクトに変換することで、データが可視化される。

4.8 システムの構成

本システムは大きく分けて、アイデアプロセッサ部、ファイル入出力部、履歴レイヤー表示部からなる (図 12)。

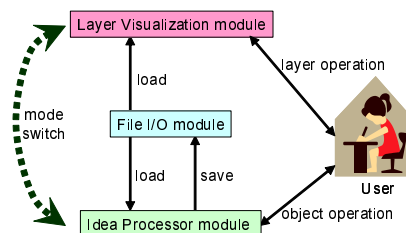


図 12: システムの構成図

アイデアプロセッサ部はツールのウインドウ上にオブジェクトを作成したり修正を加えるなど、一般的な知識創造支援ツールと同等の機能を提供する。

ファイル入出力部はアイデアプロセッサ部と密接に関連している。XML 文書からデータを読み込んでアイデアプロセッサ部に渡したり、アイデアプロセッサ部のデータを XML 文書に保存する部分である。

履歴レイヤー表示部は、履歴レイヤーの表示を変更することで過去のアイデア変遷などを振り返ったりできる機能を提供する。

ユーザは主にアイデアプロセッサ部を用いて、知識創造活動を行う。活動によって産み出されたアイデアなどはファイル入出力部によって保存・読込が行われる。蓄積された履歴を振り返りたい場合は履歴レイヤー表示部が提供する機能を用い、履歴レイヤーを操作して活動を振り返ることができる。

5 評価と考察

5.1 ツールを試用しての所見

IdeaCreppe について実際に簡単な知識創造活動を実施し、操作感などを調査した。

オブジェクトの生成や編集に関しては、ユーザはマウス操作の使い分けによって行いたい操作を容易に選択できる。多くの既存ツールがツールボタンなどによるモード切替を採用している中、本ツールはより直感的な操作で知識創造活動が行えると考えられる。

レイヤー構造によって履歴を参照できることによる利点は現在明らかにされていない。しかしながらユーザの主観的な感想としては、過去の活動を振り返ることで今後の発想の指針を決める参考になったといえる。今後は履歴を参照することが知識創造活動においてどのような効果を上げるのか、定量的な評価が望まれる。

ツールのユーザインタフェースおよび操作体系については大いに議論の余地があるといえるだろう。先行研究 [9] において、スライダを用いたレイヤー操作はユーザに対してレイヤーを操作しているという印象を与えにくいという実験結果が得られている。このインタフェースを改善することでレイヤーの概念がより分かりやすくなり、履歴レイヤー構造の有用性が増すのではないかと考えられる。

5.2 履歴レイヤー構造の提示手法改良に向けて

現在の IdeaCreppe はスライダー操作だけが履歴レイヤー構造を表現しており、ユーザはスライダーとレイヤーの概念を結びつけることが困難である。そのため、一目見ただけでレイヤーを連想させるようなものをツールに実装する必要がある。

最も簡単な方法として、履歴レイヤー表示モードではネットワーク図を表示している部分の背景にレイヤーの概念図 (図 2 がそれに近い) を表示し、スライダー操作と連動して可視化されているレイヤーに色づけしてユーザに提示するような方法が考えられる。この方法はインタフェースに変更はないものの、そのインタフェース操作がレイヤー操作に繋がっているというフィードバックをユーザに与えることができる。

その他の手法として、操作インタフェースをスライダー以外のものに置き換えるという方法がある。例えばレイヤーの概念図を直接クリックするなどして閲覧したいレイヤーをピックアップしたり、閲覧したい範囲をなぞったりして選択することができれば、ユーザはレイヤーを操作しているということが直感的に理解できると考えられる。

このように、履歴レイヤー構造の提示手法には様々な改良が考えられる。

5.3 入力デバイスの考察

現在の IdeaCreppe はマウスとキーボードによる入力を想定して作成している。しかしながら近年の計算機環境は実世界指向の流れにより、ペン (スタイラス) を用いることで我々が紙とペンを用いた実世界での活動により近い操作を行うことを可能にした。このような現状を踏まえ、創造作業支援においても手書きのメリットを生かす試みが研究されるようになってきた [8]。多くのデザイナーは高機能の計算機が存在しても、発想の初期段階においては紙とペンによる自由なデザインを好み、計算機上で行う場合も素早く自由に描くことを求めている [7]。

本研究で開発した IdeaCreppe も、マウスとキーボードによる入力だけでなく、ペンを用いた手書き入力による利用も視野に入れたいと考えている。クリックで整形されたノードを生成するのではなく、キャンバス上でフリーハンドにより矩形を描画すればノードを生成し、ノード間を線で接続すればエッジが生成されるような機能の実装が考えられるだろう。現在の IdeaCreppe はすでにモード切替を意識しない操作体系を考慮に入れており、フリーハンドの自由な入力に対応させることは十分可能であると考えている。

6 既存ツール、関連研究

近年は計算機の爆発的な普及に伴い、計算機上で電子的に知識創造支援を行う、いわゆる知識創造支援ツール⁴も数多く開発されるようになった。それらは商用のもの [1] から個人が作成したフリーウェア [2, 4]、シェアウェア [5] まで幅広く、様々な特徴で他との差別化を図っているものも多い。たとえば FreeMind [3] は Mind Maps の技法による知識創造支援に特化したツールであり、机上での活動である Mind Mapping を計算機上で実現できる。既存ツールは多くの機能を持ち実用性の高いものが多いが、その一方で過去のアイデアを記録し履歴を保存することは重視されておらず、データは修正を加えるたびに上書きされる仕様になっているツールがほとんどである。

学術的な研究としても、発想支援、知識創造支援と呼ばれる分野の研究は長い歴史を持つ。初期の研究として有名なものに Stefik ら [?] があり、これは複数人による協調作業において計算機による支援を考える研究であった。一方で、「発想支援」と呼ばれるものにはどのような形態が考えられるのかについても様々な研究・提案がなされており、折原 [10] や國藤ら [11] によってよくまとめられている。

知識創造支援ツールの開発は様々な研究者によってなされており、国内においては特に KJ 法の支援に特化したツールが多く見受けられる。三末らは

⁴発想支援ツール、アイデアプロセッサなどとも呼ばれる

KJ法の過程を図的思考展開過程としてとらえた対話型支援システム D-ABDUCTOR[12]を開発し、計算機能力を活用することで発想支援の新しい可能性を実現した。由井園らは発想支援グループウェア郡元 [13]を開発し、複数人が計算機を通して KJ法を行うシステムを提案している。

先行研究はツール上でアイデアをいかに扱うかを主眼において提案・開発がなされたものが多い。本研究はこのような先行研究とは別の着眼点、すなわち履歴保存の重要性を中心に知識創造支援を考えた。そのため本研究は機能拡張により、先行研究の様々なツールに応用することができると考えられる。

履歴を管理するという考え方は、CVS (Concurrent Versions System)[6]に通じるものがある。CVSはプログラムのソースコードなど、テキスト情報の差分を管理するシステムである。これに対し、本研究で用いた履歴レイヤー構造はネットワークの差分を可視化することに特化しており、対象となるデータが異なっている。

7 まとめ

本研究ではレイヤー構造ネットワークを知識創造支援ツールに応用し、“IdeaCrepe”を実装した。本ツールは様々な知識創造支援活動を支援する汎用的なツールであり、レイヤー構造によってイベントの履歴を保存している。この履歴を参照することで過去の発想過程を思い出すことができ、新しいアイデアを生み出す力になると考えられる。

今後の研究としては、ユーザがより直感的にレイヤー操作を行うことができるように、ネットワーク図そのものを「レイヤー」と分かるような形で描画されるように改良を加える予定である。また、スライダーを使った操作に代えてレイヤーそのものをピックアップしたりドラッグするといった操作体系を導入するために研究を継続している。

さらに、現在の整形されたオブジェクト生成ではなく、フリーハンドでより自然な入力をサポートすることにより、ユーザの発想をさらに自然に引き出すことが可能になると考えられる。

また、知識創造活動において履歴の振り返りがどのような効果をもたらすのかについては現在明らかにされていない。この効果について定量的な評価を行うことが望まれる。

参考文献

- [1] JUSTSYSTEM アイデアマスター.
<http://www.justsystem.co.jp/ideamaster/>
- [2] iEdit.
<http://homepage3.nifty.com/kondoumh/software/iedit.html>
- [3] FreeMind.
<http://freemind.sourceforge.net/>
- [4] IdeaFragment2.
<http://homepage3.nifty.com/neko33/lzh/ideafrg2.htm>
- [5] IdeaTree.
<http://www.dicre.com/soft/itree.htm>
- [6] CVS - Open Source Version Control.
<http://www.nongnu.org/cvs/>
- [7] J. A. Landay & B. A. Myers. Sketching Interfaces: Toward More Human Interface Design. *IEEE Computer*, Vol.34, No.3, pp. 56–64, 2001.
- [8] K. Misue & J. Tanaka. A Handwriting Tool to Support Creative Activities. *Proceedings of 9th International Conference on Knowledge-Based & Intelligent Information & Engineering Systems*, pp. 423–429, 2005.
- [9] N. Nakazono, K. Misue & J. Tanaka. NeL2: Network Drawing Tool for Handling Layered Structured Network Diagram. *Proceedings of Asia Pacific Symposium on Information Visualization 2006 (APVIS2006)*, pp. 109–115, 2006.
- [10] 折原良平. 発想支援システムの動向. *情報処理*, Vol.34 No.1, pp. 81–87, 1993.
- [11] 國藤進, 山下邦弘, 西本一志, 藤波努, 宮田一乗. 知識創造支援システム研究開発の動向とJAISTにおける開発の現状. 第1回知識創造支援システムシンポジウム報告書, pp. 1–10, 2004.
- [12] 三末和男, 杉山公造. 図的発想支援システム D-ABDUCTOR の開発について. *情報処理学会論文誌*, Vol.35, No.9, pp. 1739–1749, 1994.
- [13] 由井園隆也, 宗森純, 長澤庸二. カード型データベースをもつ KJ 法一貫支援グループウェアの開発と適用. *情報処理学会論文誌*, Vol.39, No.10, pp. 2914–2926, 1998.
- [14] 川喜田二郎. 発想法. 中公新書, 1967.
- [15] 高橋誠. 問題解決手法の知識. 日本経済新聞社, 1984.