

教室内創作活動のための共有・振り返り支援システムによる 創造的思考の育成

森本 竜也¹ 高田 秀志²

概要: 近年, 児童の思考力を育成するために GUI プログラミング環境を用いた授業やワークショップが開かれている。また, このような環境を用いた学習において, 創造的思考を育成するためのモデルが提唱されている。このモデルでは *Share* や *Reflect* といったプロセスが重要であるとされているが, 現状のワークショップにどのように取り入れていくかが課題である。そこで本研究では, *Share* や *Reflect* といった活動を創作活動中に行うモデルを提案し, ワークショップ中にこのモデルを達成することを支援するシステムを構築する。この支援システムを構築するにあたり, 児童はどのような *Share* や *Reflect* を行うことが, 児童にとって良い学習効果をもたらすかを明らかにする必要がある。そこで, 創作活動中に児童が描いた絵やスクリプトを他の児童と共有することのできるシステムを 3 種類構築し, それぞれを GUI プログラミング環境を用いたワークショップに適用した。また, それらのシステムを用いてどのような *Reflect* が生まれるかを明らかにするために児童の様子を観察した。

キーワード: GUI プログラミング環境, 作品共有, 創造的思考の育成

Promoting Creative Thinking with a Sharing and Reflecting System for Creative Activity in the Classroom

TATSUYA MORIMOTO¹ HIDEYUKI TAKADA²

Abstract: Recently, classes and workshops using GUI-programming-environment have been being held in elementary education. As a model for promoting creative thinking, a process consisting of *Imagine*, *Create*, *Play*, *Share* and *Reflect* is proposed. However, there is a problem to adopt *Share* and *Reflect* in the current workshop which are essential in the model. In this paper, we propose a model on which children perform *Share* and *Reflect* in creative activity, and describe the development of a system to support achieving the model in workshops. To consider developing the system, what kind of *Share* and *Reflect* performed by children will lead to producing good effect on study. To satisfy this need, we developed three kinds of systems which are based on different sharing methods to share with other children drawn pictures or constructed scripts, and made an experiment to clarify what kind of *Reflect* occurs using this system.

Keywords: GUI-programming-environment, sharing works, promoting creative thinking

1. はじめに

2002 年に行なわれた学習指導要領の改訂では, 学習内容の削減や完全学校週 5 日制の実施など大幅な変更があった。また, 特徴的な変更として「総合的な学習の時間」が

新設された。小学校学習指導要領 [1] によると, 総合的な学習の時間のねらいとして, “再生的思考”と“創造的思考”の育成が掲げられている。これらの思考力の育成として, 初等教育において Squeak Etoys や Scratch といった GUI プログラミング環境を用いた授業やワークショップが開かれている [2]。また, GUI プログラミング環境を用いた学習において, 創造的思考を育成するためのモデルとして

¹ 立命館大学大学院理工学研究科

² 立命館大学情報理工学部

Mitchel Resnick が提唱するプロセスモデルがある [3] . このモデルは , *Imagine, Create, Play, Share, Reflect* といった活動を繰り返し行うことで児童の創造的思考が育成されるというものである .

本研究におけるワークショップとは , 児童がファシリテーターやサポーターの支援を受けながら創作活動を行い , 作成した作品を発表会で他の参加者にプレゼンテーションするようなものを想定している . このようなワークショップでは *Share* や *Reflect* といった活動を行う機会が少ない . また , これらの活動を行うことができ , フィードバックを得られたとしても , それらを次の創作活動に活かすににくいといった問題がある . そのため , Mitchel が提案するモデルをそのまま現状のワークショップに取り組むことは難しい . そこで , 本論文では GUI プログラミング環境を用いた学習において , 創作活動中に *Share* や *Reflect* といった活動を行うモデルを提案し , ワorkshop中にこのモデルを達成することを支援するシステムについて述べる .

この支援システムを構築するにあたり , 児童がどのような *Share* や *Reflect* を行うことが , 児童にとって良い学習効果をもたらすかを明らかにする必要がある . そこで , 創作活動中に児童が描いた絵やスクリプトを他の児童と共有することのできるシステムを構築し , GUI プログラミング環境を用いたワークショップに適用した . システムとしては , 共有方法の異なる 3 種類を構築した .

以下 , 本論文の 2 節では , GUI プログラミング環境を用いたワークショップの現状と創造的思考の育成に関する問題点について説明する . 3 節ではワークショップに適した創造的思考育成モデルの提案について , 4 節ではそれぞれの共有システムをワークショップに適用した実践について , 5 節ではそれぞれの共有方法に関する考察について述べる . 最後に 6 節でまとめを述べる .

2. GUI プログラミング環境と創造的思考の育成

2.1 GUI プログラミング環境

2002 年に新設された「総合的な学習の時間」では , “再生的思考”と“創造的思考”の育成が掲げられている . 再生的思考とは算数や理科のように , まず定義や定理を学び , それらの知識を利用して問題解決を行う思考力のことである . 一方 , 創造的思考とは自ら課題を見つけ , 主体的に学習 , 思考 , 判断し , 問題解決を行う思考力のことである [4][5] . 現在の初等教育では , 再生的思考は算数や理科のような科目で , 創造的思考は図工のような科目で育成されている .

また , これらの思考力を統合した力を育成する学習ツールとして注目されているのが , Squeak Etoys や Scratch といった GUI プログラミング環境である . 図 1 は Squeak eToys が動作している画面のキャプチャである .

Squeak Etoys には , “すすめる”や“まわす”といった



図 1 Squeak Etoys

動作に関する命令や , 条件分岐 , 繰り返しなどのプログラムが , あらかじめタイルとして用意されている .

これらのタイルを組み合わせることでスクリプトを作成し , 児童がペンタブレットやマウスを用いて描いたスケッチに対してスクリプトを付加することで , そのスケッチに対し動きを与えることができる . このような特徴から , プログラミングを行ったことがない人やキーボードの扱いになれていない人でも , 比較的簡単にプログラミングを行えるため , 初等教育の現場で活用されている . さらにこれらの環境は , 児童がスクリプトに変更を加えると , その変更がスケッチの動きにリアルタイムかつ視覚的に反映され , 児童が試行錯誤しながら創作活動に取り組むことができるため , 創造的思考力の育成に適した環境であるといえる [6] .

2.2 創造的思考育成モデル

GUI プログラミング環境を用いた学習において , 創造的思考を育成するためのモデルが Mitchel Resnick によって提唱されている . このモデルは , *Imagine, Create, Play, Share, Reflect* といった活動を繰り返し行い , これらのプロセスを通して得られた経験を活かし , 再び創作活動を行うことによって創造的思考が育成されるといったものである .

Mitchel はこのプロセスが達成できるアプリケーションとして Scratch[7] をあげている . Scratch のアプリケーションには「共有ボタン」が用意されており , ユーザがそのボタンをクリックすることによって , 作成した作品を Scratch のホームページ上にアップロードすることができる . アップロードされた作品は , 世界中の誰でも自由に閲覧することができる . また , その作品に対して , コメントをしたり , Like ボタンを押したり , 作品をダウンロードしたりすることで , その作品を作成したユーザに対してフィードバックを返すことができる . アップロードしたユーザはその作品の再生回数や Like 数 , その作品に対してのコメントなどを得ることができる . Scratch では , *Imagine, Create, Play* は創作活動中に , *Share, Reflect* は創作活動後にホームページ上で行なわれるものとされている . ユーザはホームページ上で *Share* し , コメントやアドバイスといったフィード

バックを得る (*Reflect* する) ことで、次回の創作活動にそれらを活かすことができ、作品の創作スキルや、創造的思考が向上されるといわれている。

2.3 GUI プログラミング環境を用いたワークショップにおける問題点

2.2 節で述べた Mitchel のモデルは児童の創造的思考の育成に適しているといわれているが、GUI プログラミング環境を用いたワークショップにこのモデルをそのまま適用するためには、*Share* や *Reflect* といったプロセスにおいていくつかの問題がある。まず、児童が他の児童と作品を共有するためには、創作活動中に自席から離れ、他の児童の PC の側まで行き作品を閲覧しなければならない。また、ワークショップで他の児童の作品を閲覧する機会として、ワークショップの最後に行われる発表会がある。しかし、発表会では発表者の意向でプレゼンテーションが進められてしまうため、気になったり参考にしたいスケッチやスクリプトがあったとしても、それらをじっくり見ることができない。さらに発表会では、発表者と聴衆という役割が必然的に与えられてしまうため、児童同士で意見交換を行うといった活動が行われにくく、フィードバックを得る機会が少ないといえる。また、一般的な授業で GUI プログラミング環境を用いる場合は、複数回にわけて一つの作品を作り上げる創作形式を取ることが多い。しかし、GUI プログラミング環境を用いたワークショップでは 1 回につき 1 つの作品を完成させる必要がある。そのため、創作活動後に他の児童と作品を共有し、フィードバックを得ることができたとしても、次の創作活動にそのフィードバックを生かすことができない可能性がある。

Mitchel は Scratch の「共有機能」や「コメント機能」を用いることで、*Share* や *Reflect* といった活動を行うことができる」と述べている。しかし、現在 WEB 上にアップロードされている作品は約 270 万件にもおよび、その中からワークショップの参加者の作品を探すのは困難である。さらに Scratch のコメント機能による *Reflect* はキーボード入力を必要としており、キーボード操作に慣れていない児童に利用してもらうことは難しい。そのうえ、ワークショップは児童同士で直接会話が行える環境であるのに対し、コメント機能によってフィードバックを返すということは、会話によるコミュニケーションを減少させてしまう恐れがあり、その環境を活かしきれていないといえる。

3. 創作活動中における共有・振り返り支援

3.1 スパイラル&タックモデル

2.3 節で述べたように、GUI プログラミング環境を用いたワークショップに Mitchel のモデルを適用しようとする、創作活動と共有や振り返りの時間が分断されてしまうため、得られたフィードバックを作品に生かすことができ

ないといった問題がある。そこで、図 2 で示されるように、*Share* や *Reflect* を創作活動中に行うような「スパイラル&タックモデル」を提案する。

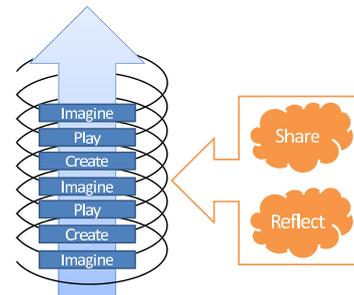


図 2 スパイラル&タックモデル

現状のワークショップでの創作活動は、*Imagine*、*Create*、*Play* といった 3 つの活動がらせん状に繰り返されている。これらの活動が行われたあと、*Share* や *Reflect* が行なわれているため、創作活動のプロセスと分断されてしまっている。スパイラル&タックモデルは、現状のワークショップでは創作活動終了後に単独で行われていた *Share* や *Reflect* を創作活動中に取り込んだモデルである。このモデルに従いワークショップを進行することで、創作活動と並行して他の児童と作品を共有し、フィードバックを得ることが可能となり、そこで得られた新しいアイデアやコメントを創作中の作品へと反映することができる。

3.2 創作活動における *Share* と *Reflect*

スパイラル&タックモデルをワークショップに適用するにあたり、GUI プログラミング環境を用いたワークショップにおいて、どのような *Share* や *Reflect* を行うことが児童の創造的思考の育成に対し良い影響を与えるのかを検討する必要がある。教室でのワークショップにおいて、GUI プログラミング環境を用いて作成された作品を共有する方法として、“作業端末上での共有”、“スクリーンを用いた共有”、“小型端末を用いた共有”などが挙げられる。また、ワークショップにおいて *Reflect* とは、児童同士で直接会話することでコメントやアドバイスをお互いに行うことなどが挙げられる。これまでワークショップの創作活動中に *Share* といった活動を行なわれてこなかったため、そのような活動を行ったときに、児童間でどのような *Reflect* が起こるのかが明らかになっていない。そこで、上述した作品共有方法について、それぞれの方法をワークショップに適用することで児童の創作活動にどのように影響を与えるかを分析するために、共有方法の異なる 3 種類の作品共有システムを構築し、実際のワークショップで児童に利用してもらった。また、それぞれのシステムを利用することで、児童はどのような *Reflect* 活動を行うのかを観察した。

4. 実践

4.1 作業端末上での共有

創作活動中に児童が描いたスケッチや構築したスクリプトを各児童の作業端末上で共有するシステムを構築し、ワークショップで児童に利用してもらった。

4.1.1 システム概要

本システムでは、児童がワークショップ参加者にスケッチやスクリプトを共有したいと考えたとき、「共有ボタン」を押すことで、他の参加者の端末にポップアップによって作品が表示される。ポップアップはある一定の時間が経つと次第に消えるようになっている。

本システムの動作画面を図3に示す。今回は「共有ボタン」を Squeak Etoys のアプリケーション上に実装した。このボタンがクリックされると、選択されたスケッチやスクリプトがサーバにアップロードされる。ワークショップで使用される PC にはあらかじめサーバのフォルダがマウントされており、そのフォルダ内に新たなファイルが書き込まれるとポップアップ表示システムに通知される。ポップアップ表示システムは常駐プログラムとして動作しており、通知がある度にスケッチやスクリプトをポップアップで表示する。ポップアップは3秒間表示されると徐々に透明度が高くなり、最後には完全に透明になることで児童には見えなくなる。また、ポップアップにマウスオーバーされると透明度が低くなり、作品を表示し続けることが可能となる機能を持つ。



図3 作業端末上での共有システム

4.1.2 実践内容

4.1.1 節で述べたシステムを、2012年7月8日に Squeak Etoys を使用したワークショップに適用した。ワークショップの参加者は11人で、学年は1年生から4年生であった。また、11人中10人の児童はこのワークショップで初めて Squeak Etoys を利用した。児童は「レーシングゲームをつくらう」というテーマに沿って、Squeak Etoys を用いて創作活動を行った。今回は、共有する方法のみを児童に伝え、児童が共有したいタイミングで自由に共有してもらった。システムは、児童の共有回数、参照回数を取得した。

共有回数は児童が Squeak Etoys の「共有ボタン」を押した回数を、参照回数はポップアップに3秒以上マウスオーバーした回数を計測した。また、作品が共有されたときの児童の反応を観察した。

4.1.3 観察結果

児童がアップロードした回数を計測した結果、合計で44回であった。また、児童がアップロードされた作品を参照した回数を計測した結果、合計で29回であった。参照した時間を観測すると、3~5秒閲覧している児童が大半であったが、1人の児童はある作品に対し44秒間閲覧している様子が観察された。また、共有活動を児童が行ったときの児童の反応を観察した結果、

- 見て！！これは僕が描いた絵！！(アピール)
- これは何のつもりで描いたの？(質問)
- いいやん！！(評価)
- もっと色を使ってみれば？(アドバイス)

という会話が行われていた。しかし、会話が行われていたのは隣同士に座っている児童のみであった。さらに、スケッチをアップロードした児童が隣の児童に話しかけるといった活動をしなければ、会話が生まれなかった。

4.2 スクリーンを用いた共有

創作活動中に児童が描いたスケッチや構築したスクリプトを教室の前に設置したスクリーンを用いて共有するシステムを構築し、ワークショップで児童に利用してもらった。

4.2.1 システム概要

本システムでは、児童がワークショップ参加者にスケッチやスクリプトを共有したいと考えたとき、「共有ボタン」を押すことで、教室に設置されたスクリーンに作品が表示される。スクリーンには新しく共有された順に複数のスケッチやスクリプトが表示されるようになっている。4.1.1 節で述べたシステム同様、Squeak Etoys のアプリケーション上に「共有ボタン」を実装した。このボタンがクリックされると、サーバに児童が選択したスケッチやスクリプトがアップロードされる。スクリーン投影用 PC には、サーバの共有フォルダがあらかじめマウントされており、このフォルダに新たなファイルが書き込まれるとスクリーン表示システムに通知する。スクリーンには8つのスケッチが表示され、古いものから削除される。

4.2.2 実践内容

4.2.1 節で述べたシステムを、2012年10月14日に Squeak Etoys を使用したワークショップに適用した。ワークショップの参加者は10人で、学年は小学4年生から中学2年生であった。また、10人中9人の児童はこのワークショップで初めて Squeak Etoys を利用した。児童は「射的ゲームをつくらう」というテーマに沿って、Squeak Etoys を用いて創作活動を行った。今回も、共有する方法のみを児童に伝え、児童が共有したいタイミングで自由に共有しても

らった．システムは，児童の共有回数を取得した．共有回数は児童が Squeak Etoys の「共有ボタン」を押した回数を計測した．スクリーンに表示された作品を児童がいつ見たのか，何回見たのかは計測していない．また，作品が共有されたときの児童の反応を観測した．

4.2.3 観察結果

児童がアップロードした回数を計測した結果，合計で 19 回であった．2 人の児童が積極的に作品をアップロードしており，それぞれ 6 回と 7 回であった．共有を一度もしていない児童も存在した．スクリーンに表示された作品を閲覧している児童もいたが，多くの児童は創作活動に集中しており，そもそもスクリーンに新たな作品が表示されたことに気づいていない児童がいた．また，スクリーンを用いて作品を共有したことによる児童間での会話は生まれなかった．

4.3 小型端末を用いた共有

創作活動中に児童が描いたスケッチや構築したスクリプトを，児童が使用している PC と連携して動作する小型端末 (Galaxy Nexus) 上で共有するシステムを構築し，ワークショップで児童に利用してもらった．

4.3.1 システム概要

本システムでは，児童がワークショップ参加者にスケッチを共有したいと考えたとき，小型端末を振ることで他の参加者の小型端末にスケッチを送信することができる．受信したスケッチは端末に保持されており，児童はどのタイミングでも過去の共有作品を閲覧することができる．端末は他の児童から作品を受信すると，端末利用者に通知する機能を持つ．

システムを起動すると，児童のニックネームを入力するフォームが現れる．ここで入力されたニックネームがアップロード用フォルダの名前となる．今回のシステムを実現するために拡張された Squeak Etoys は，1 分間に 1 度プロジェクト内の全モーフを取得し，サーバの個人用アップロードフォルダにアップロードする．小型端末上では作品交換アプリケーションを実行する．アプリケーション起動時に，サーバのフォルダ名を全て読み込み，リスト形式で提示する．ユーザに選択されたニックネームを元に，小型端末はサーバから作品を全て読み込み，それらを図 4 のようにグリッド形式で表示する．

表示された作品をタッチすると，その作品を他の端末へ送信する画面に遷移する．この画面で端末を振ると，ブロードキャスト送信によって，作品が全端末に送信される．メッセージの内容はフォルダ名とファイル名であり，受信した端末はそれらを元に作品をサーバよりダウンロードする．受信端末がメッセージを受信すると，音による通知を行う．これにより，ユーザが端末に触れていないときでも，新たな作品を受信したことを通知することができる．

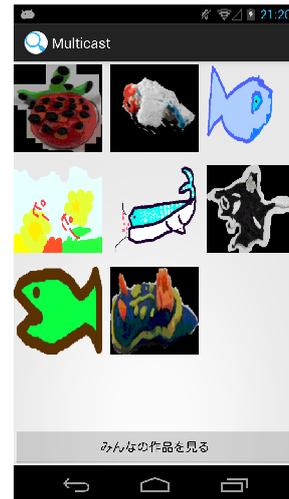


図 4 小型端末を用いた共有システム

また，受信した端末には全ての作品が保持されており，過去に共有された作品を全て閲覧することができる．

4.3.2 実践内容

4.3.1 節で述べたシステムを，2012 年 12 月 2 日に Squeak Etoys を使用したワークショップに適用した．ワークショップの参加者は 5 人で，学年は小学 3 年生から小学 5 年生であった．また，5 人全員が Squeak Etoys を使用したワークショップに参加したことがあり，Squeak Etoys の操作には慣れていない．児童は「アニメーションで動く図鑑をつくろう」というテーマに沿って，Squeak Etoys を用いて創作活動を行った．今回は，共有する方法のみを児童に伝え，児童が共有したいタイミングで自由に共有してもらった．システムは，児童の共有回数を取得した．共有回数は受信した作品数を計測した．今回は，児童が何回作品を参照したかは取得できなかった (システムの不具合による)．また，作品が共有されたときの児童の反応を観測した．

4.3.3 観察結果

共有回数は 55 回であった．しかし，同じ作品が何度も共有されていたことがわかった．児童は端末を振り，自身の作品が他の端末に送信されるという仕組みにおもしろみを感じ，何度も同じ作品を共有していたようにみえた．また，作品が共有されたとき，その作品を見た児童が共有した児童に対して，「これ，何の絵？」といった質問をしていた．多くの児童は作品が共有されたときには確認せず，後でまとめて閲覧をしていた．しかし，まとめて閲覧をしているときは，質問やコメントなどの会話は生まれていなかった．

5. 考察

今回，共有方法の異なる作品共有システムをワークショップに適用した結果から，それぞれの共有方法について以下の観点から考察した．

- 共有活動を行うときの児童の動機について
- 共有活動とそれによる児童の反応との関係性について
- 共有活動を行うことで発生した会話となぜそれらが発生したのかについて
- *Share* や *Reflect* を増加させるために必要となる機能について

5.1 作業端末上での共有

1人あたり平均4回の共有活動が行われたことは、我々が予想していた回数よりも多かった。児童は他の児童のPC上に自身の作品が表示されることに興味を持ち、その様子を楽しんでいる光景が見られた。このことから、ポップアップによる作品共有方法は、児童が積極的に共有活動を行うなど、ある程度の効果が期待できることがわかった。会話が生まれると、お互いの作品への評価や、その作品をどういった考えで描いたかといった質問、アドバイスやコメントといった創作活動において有効といえる会話がされていたことがわかる。

しかし、隣同士の児童しか会話が起こらなかったことや、アップロードされた作品を児童がアピールしなければ会話が起こらなかったことから、作品をポップアップ表示させるだけでは *Reflect* の支援にはならないことがわかった。また、今回使用した作業端末上での共有システムでは、誰が作品をアップロードしたかを明示しなかった。そのため、アップロードされた作品に対して、児童が質問や評価をしたくても、誰に対して行えばよいかわからずに *Reflect* が行われなかったと考えられる。また、ポップアップが表示される領域で児童が作業を行なっているとき、急にポップアップが表示されることで児童の作業が中断される様子が観察された。一方で、ポップアップが表示される領域外で児童が作業を行なっているときは、児童は特に気にすることなく作業を進めることができていた。このことから、作業スペースと共有スペースを分離することで、児童の創作活動を中断させることなく、*Share* することが可能になると考えられる。

5.2 スクリーンを用いた共有

スクリーンを用いた共有方法では、スクリーンを使用することで児童が同じ領域で共有することができ、それにより教室全体で意見交換が行なわれることが期待された。しかし、実際は共有回数が少なく、会話も発生しなかった。これは、作品が共有されスクリーンに新たな作品が表示されたとき、そのことに気付かずに創作活動を行なっている児童が多かったからであると考えられる。*Share* や *Reflect* を増やすためには、児童に作品が共有されたことを通知し、その作品に興味を向けることが必要であると考えられる。一方で、共有回数が多い児童もいた。これらの児童は、スクリーンに自身の作品が表示され続けていることに喜びを

感じ、共有活動を行なっていた。このことから、長時間作品が表示されることが共有のきっかけになっていることが予想される。

5.3 小型端末を用いた共有

小型端末を用いた共有方法では、共有回数は多いが、同じスケッチばかりが共有されるといった結果になった。これは、端末を振って作品共有すること自体に楽しみを感じ、その動作を繰り返し行なったことで、共有回数が増加したと考えられる。今回の実践では、振る動作によって作品を共有することで、教室全体が賑やかになり、共有回数が増え、会話も増えるだろうと予想していたが、実際は共有回数は多かったにも関わらず、あまり会話が行なわれなかった。また、音を利用した通知によって、作品が共有されるたびに児童が作品を閲覧し、会話が行なわれることが期待された。しかし実際は、音による通知のみでは、児童は作品を閲覧せず、創作活動を続けることがわかった。また、通知を受けたときではなく、小型端末に蓄積された作品をまとめて閲覧するといった児童がいたことから、過去に共有された作品を閲覧することのできる機能は提供すべきであることがわかった。しかし、過去に共有された作品を児童が閲覧しても、会話は発生しにくいいため、リアルタイムに作品を閲覧してもらい、補足として過去の作品を閲覧できる機能として提供すべきであると考えられる。また、同じスケッチが何度も共有されたことから、同じスケッチは一定時間送信できないなどの制限を設けることで、不用意なスケッチの共有は防ぐ必要がある。

5.4 全実践を通じての考察

- 共有活動に対する児童の動機
児童には様々な方法で共有作業を行なってもらったが、共通していることとして、児童が作品を共有するとき、完成した作品のみを共有していた。スケッチを共有したあとに、そのスケッチを書きなおすといった修正を加えた児童は一人もいなかった。スパイラル&タックモデルにしたがってワークショップを進めることのメリットとして、創作活動中でも意見をもらうことができ、その場で作品に反映させることができるということである。しかし、3回の実践を通してこのような活動を行った児童はいなかった。児童は「こんなスケッチが描けたから見て欲しい」といったモチベーションで作品を共有していたと考えられる。
- 送信方法による共有活動の違い
今回実践を行った3種類の共有方法のなかで、最も共有回数が多かったのが、小型端末を利用した共有方法である。他の2種類の共有方法では、*Squeak Etoys* に拡張された共有ボタンをクリックすることで作品を他の児童の端末に送信していた。一方で小型端末を利用

した送信方法では、端末を振ることで作品を他の児童に送信していた。このように、児童は体を動かすことで作品が送信されることがおもしろいと感じ、共有回数が増えていたと考えられる。共有回数が増えることで *Reflect* の機会の増加に繋がることが見込まれるため、作品の送信方法としては小型端末を利用した動きのある送信方法が最良であると考えられる。

- 表示方法による会話の発生の違い

今回は3種類の表示方法を実践で使用したが、スクリーンや小型端末を利用した共有方法では、あまり会話が発生しなかった。一方で、作業端末上での共有方法では、効果的と思われる会話が発生していた。スクリーンや小型端末を利用した共有方法では、新たな作品が共有されたとき、児童が気付かなかったり、その時には確認せずにあとで作品をまとめて閲覧したりと、その場で確認するという作業が行なわれなかった。作業端末上での共有方法では、作業領域に共有された作品が表示されるため、創作活動に集中している児童の目に止まりやすく、会話による *Reflect* が行われたと推測できる。共有された作品の表示方法としては、作業端末上での共有方法のように、創作活動中の児童にある程度強制的に通知する仕組みが必要であると考えられる。また、スクリーンに表示されている作品を見て喜んでいる児童や、小型端末に蓄積された作品を後で見返している児童がいたことから、共有されたときだけでなく、後からでも作品を参照することのできる機能が必要である。

今回の実践を通じて、作品の送信方法や表示方法について、児童に最適であると思われる機能要件を確認することができたため、今後はこれらを組み合わせたシステムの構築を行っていく。

- *Reflect* の発生

今回の実践では、会話による *Reflect* があまり行なわれなかった。そのため、現状では児童にフィードバックが返ってきていない状況が多々あった。そこで、SNSなどでも用いられている「いいねボタン」のような、簡単な操作でフィードバックを返す *Reflect* 機能を持たせることで、作品を共有した児童に何らかの反応を返す仕組みが必要となる。これらの機能を持たせることで、共有活動にどのような影響を及ぼすのかを明らかにしていく。

6. おわりに

本論文では、創作活動中に *Share* や *Reflect* を行う「スパイラル&タックモデル」を提案した。このモデルを達成することを支援するシステムを構築するために、児童がどのような *Share* や *Reflect* を創作活動中に行うことが、児童の創作活動に良い影響を及ぼすのかを明らかにする必要

があった。そこで、スケッチやスクリプトを共有することのできる共有システムを構築し、GUIプログラミング環境を用いたワークショップに適用した。

その結果、作品の送信方法や表示方法によって、*Share* や *Reflect* に対する児童の反応が異なることがわかった。それぞれの実践結果から、作品の送信方法や表示方法において、児童に最適と思われる機能の要件を得ることができたため、今後は、これらを元にスパイラル&タックモデルを達成することを支援するシステムの構築を行っていく予定である。

参考文献

- [1] 小学校学習指導要領/第5章総合的な学習の時間, http://www.mext.go.jp/a_menu/shotou/new-cs/youryou/syo/sougou.htm
- [2] 荒木貴之: ロボットが教室にやってくる-知的好奇心はこうして伸ばせ- 立命館小学校のアイデア, 教育出版, 2008.
- [3] Mitchel Resnick: All I Really Need to Know(About Creative Thinking) I Learned(By Studying How Children Learn) in Kindergarten, Creativity & Cognition conference, 2007.
- [4] 創造性に関する一研究, 玉城政光, 日本教育方法学会紀要 14, 11-19, 1989.
- [5] 創造的思考を育成するためのコンテクスト創造的協調学習支援システムに関する研究, 稲葉光行ら, アートリサーチ 4, 165-178, 2004.
- [6] 大座畑 重光, Squeak における Morphic プログラミング, ヒューマンインタフェース研究会報告, 17-24, 2005.
- [7] John Maloney, Leo Burd, Yasmin Kafai, Natalie Rusk, Brian Silverman, Mitchel Resnick.: Scratch: A sneak preview. Creating, Connecting and Collaborating through Computing, 2004. pp.104-109, 2004.