Stories Children write while Coding: a cross-disciplinary approach for the Primary classroom

Journal:	*Cambridge Journal of Education*
Manuscript ID	CCJE-2017-0086.R1
Manuscript Type:	Original Article
Keywords:	story writing, primary literacy, computer programming, coding

Stories Children write while Coding: a cross-disciplinary approach for the Primary Classroom

Abstract

This paper presents research on how children aged 5-11 can create stories through computer programming (coding). We introduce a novel 'Story-Writing-Coding' engine where children realise their imagined stories through code that results in animations representing their stories. Analysis of how they manipulated code to represent story meaning is presented, together with a hypothetical model of the strategies they used in composition. Using the context of multimodal literacy we argue that coding can be thought of as a form of literacy and that taking this perspective may lead to benefits for children as both literacy learners and also learners of computer programming. While we have worked with ~~over~~ 103~~100~~ children, the results presented in this paper are drawn from an in-depth analysis of 20 stories with associated code together with observations by the authors. We find children are able to manipulate code creatively to obtain meaningful stories and successfully use the modes of print, static and moving images.

Keywords: story writing, primary literacy, new literacies, multimodality, Systemic Functional Theory, story writing, computer programming, coding.

Subject classification codes: include these here if the journal requires them

Introduction

The effect of digital technology and applications has a wide-ranging impact on the everyday lives of our young children, and this is advancing. Even our youngest children experience a range of digital devices, PCs, tablets, smart-phones (Alexander, 2010). Many play in virtual worlds such as *Minecraft* and *Club Penguin* watch YouTube videos, use the Internet or play with apps on a tablet (Marsh, 2010; Marsh et al., 2015). The educational value of apps has been tested; Marsh et al. (2015) find that characteristics such as problem-solving, critical thinking and reasoning facilitate children's creativity, factors which are also highlighted by Burnett, Davies, Merchant and Rowsell (2014). However, simply using an app can situate children as consumers rather than creators of digital media, while activities which apply digital technologies such as filmmaking and digital puppetry can shift the focus to the creative (Wohlwend, 2015; Husbye, Buchholz, Coggin, Wessel-Powell and Wohlwend, 2012).

A shift from consumers to creators of digital technology should provide children with experiences so they become 'digitally literate', where technology is used to 'establish a social practice including a mix of reading, writing and meaning creation using other modes of expression ' (Sefton-Green, Marsh, Erstad and Flewitt, 2016). Over a decade ago Mitchel Resnick, MIT Media Lab's Professor of Learning Research, coined the term 'digital fluency', the ability to be creative with technology, and foresaw computers as 'a new medium through which people can create and express' (Resnick, 2002).

More recently Resnick, reflecting on computer programming (coding), suggests that coding is an "extension of writing", supporting the writing of "new types of things" including stories, games and animations (Resnick, 2013). This clear suggestion that coding is a new form of 'literacy' is supported by Cathy Burnett in her recent report "The Digital Age and its implication for Learning and Teaching in the Primary School" (Burnett, 2016). She calls us to "rethink the nature of language and literacy provision in a digital age" and situates coding alongside "new literacies" in her report. She suggests that research in the new literacies can aid the development of creative, critical and aesthetic dimensions to the Computing curriculum (Burnett, 2016, p33). We agree but also propose the inverse, that creative coding can support our children as literacy learners, including traditional print-based media, and contribute to the English curriculum. Considering coding as a form of literacy should also broaden the understanding of how meaning is created in multimodal texts (such as code, print text, static and moving images) and also how to develop classroom experiences to support conventional and new literacies, especially multimodal literacy (Walsh, 2009).

We feel that the recent introduction of Computing into the Primary National Curriculum (including coding) in England should be viewed in this context (Department for Education (DfE), 2013). Our experience as computer science educators warns that teaching coding can all too easily become abstract or mathematical. We believe that seeing coding as form of literacy could avoid this pitfall and moreover challenge and enhance the curriculum and foster reflection on the classroom practice of learning to code.

This paper presents the results of a year-long study where we have taught Primary School children aged 5-11 to creatively code from a literacy perspective. Specifically we have created a 'Story-Writing-Coding' (SWC) engine where children compose stories through the medium of writing computer code as text within a multimodal setting. Following a brief overview of how children use the engine we present an analysis of (i) how the children have manipulated code to write stories and (ii) the strategies they employed as authors, including several examples of stories they have created with associated code.

Research Approach

At the time of writing we have worked with 103 children in Years 1-6, 39 from two small rural schools and 34 from two large inner-city schools, one of which is situated in a deprived area. This was done in the context of school clubs or visits to the university. In addition we worked with 36 children at a public event hosted by our city library. The age ranges of the children are; Yrs 1&2 (11 boys, 17 girls), Yrs 3&4 (18 boys, 20 girls) and Yrs 5&6 (21 boys, 16 girls). We provided close to an authentic classroom setting for all activities.

Some 100 children were involved in the research; the results and analysis presented in this paper are based on a corpus of 20 children's stories. A mixed-mode approach to data-collection was undertaken, we collected children's code, their narrative text (written and/or read aloud), and made observations of their 'story-writing-coding' (SWC) activities in a classroom setting. In particular we observed children coding and any associated thinking aloud or conversation with others, we asked them about their SWC strategies, and we asked their teachers about their progress and to critique their work. The analysis of coded stories involved viewing code and story side-by-side, and looking for common ways in which story meaning was expressed through code. This was done by both researchers independently, uncertainties were resolved by discussion. Finally we worked intensively in dialogue with one child to critique and refine the engine. These approaches are detailed below. Children from two local primary schools, accompanied by their teachers were involved together with others from public 'coding events', accompanied by their parents. Both teachers and parents monitored the children's learning and also their activities as required by Ethical approval which was granted from the university. Participants attended two sessions, each about 1 hour in duration. The first session involved learning the basic code and engine working, the second session focussed in story-writing-coding (SWC). Either teachers or parents monitored the children's learning and also their activities as required by Ethical approval which was granted from the university

We were interested in two questions, first how children used code to achieve the meaning they wished to express in their stories, second the strategies they used in composition across the modes. Code and narrative text data were analysed based on

Systemic Functional Theory, a generalisation of Halliday's Systemic Functional Linguistics (Halliday, 2004). To discover SWC strategies, we observed the children, asked them to explain how they were working and reviewed the logs of their code changes.

This project emerged out of a prior trial intended to investigate whether children could code an animated movie using a professional language (Java). Early in the trial we observed an emergent phenomenon, many children started spontaneously to 'tell stories aloud' while coding and explained their code to us through storytelling. Our research focus therefore shifted to SWC and at this point we worked intensively with a single 8 year-old child to refine the engine.

The engine was created by the authors, one is a computer scientist with an interest in linguistics; the other is a Primary English educator with knowledge of programming.

Orientation

Before we present a detailed discussion of our findings, we provide an orienting overview of the SWC engine structure and an example of a spontaneous story. Figure 1 shows the engine components, the 'location' of the developing story and an overlay of the modes associated with each component. These include two print-based modes and static and moving image modes. As we discuss later, children were given no story prompts, following some time learning the engine they were simply asked to 'code a story of your choice'.

One or more lines of code are input into component **C** (an on-screen text-box), then on pressing a 'run' button a graphical representation of the code appears on a canvas frame as an animation **A.** As discussed below children quickly engaged with the engine and developed various strategies of writing **C** and reading **A** to develop their story **S** held in mind. At the end of the activity we asked children to tell their stories **S'** which were recorded and transcribed, though most actively shared their stories verbally with each other during the composition process.

-- Figure 1 about here –

Consider the transcript of a recorded spontaneous story told by a five year old girl 'Mia' as she was learning to code, presented alongside the code she wrote, (Table 1). Even without explanation of the code semantics, it is straightforward to understand the meaning expressed through the code statements.

We observed Mia writing the first two lines of code and then she said (1). After a short pause she said (2) and repeated this to herself and then repeated both (1 and 2) to a classmate; clearly this was important for her. Following some chat with classmates, she the coded (3) and almost immediately (4) and again repeated both (3 and 4). Finally after a longer pause she said (5) and she smiled and looked around like a satisfied author. Mia made one correction to her code during (1); the entire observation lasted around 20 minutes.

-- Table 1 about here --

Overview of the Engine and its Affordances

The engine as presented on-screen is shown in Figure 2, the left text-box **C** receives the typed code which when executed (by pressing the 'play' button) appears in the animation window **A** on the right.

--- Figure 2 about here ---

Turning now to the engine's semantics, this was inspired by our understanding of Halliday's *ideational* metafunction from his Systemic Functional Linguistics. The code vocabulary available to the writers is shown in Table 2 categorised according to a subset of Halliday's *processes*, see Chapter 5 in Halliday (2004), shown as table headings with examples of code statements available to the writers.

-- Table 2 about here –

At the bottom of the table is an example of a single code statement (bold) which instructs Grog to fly to the scarecrow. This and all other *process* code (except the *Existential*) follows the indicated participant-process-circumstance template. Circumstances in the clause are coded as 'parameters' which are written inside the brackets (). These are typically circumstance of location (on-screen location) or circumstance of 'quality', (height of a jump, speed of a spin).

The importance of using 'everyday words' is recognised in apps designed for young children (Jewitt and Adamson, 2003). Each action word in the engine's vocabulary has a suggestive and unambiguous meaning, and this corresponds exactly to what it expresses in the animation **A**. These words are in the imperative mood, creating a mental model of giving characters instructions akin to play-scripts. Similarly the names used for characters, scenery and props should be familiar to young children.

Our use of 'print-based' coding (using text) differs from other programming apps such as MIT's 'Scratch' developed by Resnick for children ages 8-16. Scratch uses an iconic drag and drop approach to assemble instructions rather than the text-based approach we advocate. We made this design choice for two reasons. First we wished to test the feasibility of using a professional coding language rather than a simplified language to be extended later; when children learn to read they do not use simplified language.

Second we wished to test the 'centrality of written representation' in a multimodal context as suggested by Merchant (2007). The use of code text is central to SWC during the composition process, but its status does become relegated during the reading of the story (**A, S'**). To our knowledge this is a novel way of using text in a multi-modal context.

Analysis of the Stories Children Coded

Here we theorise how children have used code to express the meaning in their stories **S**, looking at both the narrative text **S'** and associated code **C** together (Figure 1). Before we present our analysis let us first consider one story from the corpus, *The Rain Dance* shown in Table 3. This story composed by a 10 year-old is short, but has much of the classic Proppian structure of disruption and restoration of equilibrium**.** It starts with a statement of a problematic situation (clause 1) and a solution plan (clauses 2,3) which are not coded. The execution of the solution (clause 4) is expressed by two blocks of 6 statements, one for each character. Here from the combination of actions (jumping, spinning and moving around) we are able to *infer* the existence of the dance, which is indeed perceived viewing the animation. The consequence of their dance (clause 6) is coded as the addition of four clouds in the sky, while clause 5 introduces the mental state of the characters providing story cohesion. Finally clauses 7 and 8 wrap up the story and give a reason for the characters feeling happy. The author has distributed meaning between clauses which can be coded and which cannot; we refer to the latter as *embellishments.* Interestingly, additional meaning appears in the animation **A** where the dance is revealed as a symmetrical and synchronised action sequence, each character mirrors the movements of the other. Clearly the author has carefully thought through the design of her story and has worked hard using the affordances of **C**, **S'** and **A** to obtain an animation of what a rain dance means to her. She demonstrates an appreciation of the meaning-making potential of different modes (Walsh, 2009).

-- Table 3 about here –

To theorise how meaning in **S** is expressed through code we analysed the corpus of 20 stories using Systemic Functional Theory. Originally developed by Halliday in linguistics, this theory has been applied to other modes such as images (Kress and van Leeuwen, 2006), film and architectural space (Unsworth, 2008). The resulting system network (sysNet) is presented in Figure 3; space allows only a moderate level of delicacy to be included in this discussion. Numbers in brackets indicate the use of each system within the corpus, expressed as a percentage of the total clause count.

--- Figure 3 about here –

Reading from left to right this diagram shows the system choices made by the writers in expressing meaning in **S** through the medium of **C**. The first is the choice we have already seen, whether a clause can be coded ('codable') or not ('embellishments'). The system of codable clauses is further refined into four additional systems. The choices made within these systems can be arranged along a cline from pure code to pure story-text. Consider the story, *The Rescue*, shown in Table 4 written by a 10 year-old girl telling of the rescue of an injured Robin from a burning tree.

Clauses 1 to 4 set the scene of the story and are coded as statements C and D. A complication is presented in clauses 6 and 7 (statements A and B) then the resolution in clauses 8 to 15 (statements E,F,G,H) and finally the coda in clauses 16 and 17.

Clause 14 is *directly* coded where a single action statement corresponds to the process ‘flies’ in the clause. This is the easiest way to map meaning onto code, and is often used by the youngest writers (together with embellishments). Next along the cline is meaning *derived* from code, in clause 3 the existence of Pip (‘was’) on the hillside is *derived* from code C which adds pip there and keeps her at rest. Further along we have the *inferred* system, e.g. the meaning in clause 2 is *inferred* from several lines of code (D), and also in clause 11 where meaning is *inferred* from code F. The act of Robin’s dismounting (clause 15) is *inferred* from code H where Robin walks away while Grog stays put. Similar *inference* is seen in *The Rain Dance* where multiple lines of code are used to *infer* the action of a ‘dance’.

Next along the cline is *implication* where there is no code associated with the clause, clausal meaning is *implied* from code associated with another clause. The meaning in clause 1 is *implied* from code associated with clause 2; Grog’s jumping *implies* he ‘… was having a lovely day’. Meaning in clause 12 is *implied* from Grog moving to Robin (F) followed by both moving to the same location (G).

Finally *embellishments* have been used to convey direct speech (clauses 13,16,17) and mental emotion (clauses 4,10) and purpose (clause 9). This is representative of *embellishments* used through the corpus.

Returning to Figure 3 we see how all twenty authors have used these various systems. Unsurprisingly, only 5% of clauses used *implied* mapping since this is the most sophisticated mapping. Around a quarter used both *inferred* and *derived* mapping and a much smaller percentage used *direct* mapping. This, together with the use of embellishments, suggests that authors strived to express meaning beyond that expressed through simple action. SWC appears not to constrain authors to composing trivial action-based stories.

The Process of Story-Writing-Coding

Traditional composition has been subject to decades of research which aims to understand the process of composition including formation of story ideas and strategies to manage the composition process. These cognitive models view writing as 'problem-solving', and propose both knowledge storage and retrieval mechanisms as well as discrete processes involved in writing (Bereiter and Scardamalia, 1987; Hayes and Flowers, 1980; Hayes, 1996). There are common poles to these models, e.g., in the 'knowledge transforming' model (Bereiter and Scardamalia, 1987) experts juggle both content and rhetorical goals in their writing; while in the 'knowledge telling' model used by novice, content is generated through association of ideas, one prompting the next. Writing strategies associated with these poles can be referred to as 'outline-first' and 'interactive' (Hayes, 2006). In the outline first strategy writers are encouraged to collect story ideas in mind or as draft notes before writing commences. In the interactive or 'free-writing' strategy, writers write down ideas as they flow into mind and then review and polish their text. This strategy has been advocated by Elbow (1981, 1998); here writing is not seen as a transcription of ideas already in mind into text, but involves creating ideas in the process of writing, a 'knowledge constituting process' (Galbraith, 1999).

Composition Strategies

Formatted: Heading 2

Through talking to the children we were able to obtain some indication of the strategies they thought they were using, though this was not always easy. Nevertheless several strategy types emerged: 7 used the outline first, 9 the interactive (though three of these said a plan started to emerge half way through the process). Three of the younger writers (age 5) could not articulate their strategy, observation of their activities and inspection of their code and narrative text suggests these used the interactive strategy. Observation of one author suggested that he was storying his code, writing a story on completion of coding. This is confirmed by his very high use of embellishments (69%) in his narrative text and this story was excluded from the analysis.

Based on our observations and the above data we propose the model shown in Figure 4. Central to this are two feedback loops. In the inner loop authors write their code and read the resulting animation **A** and stay in this loop, modifying their code, until **A** matches their current story idea **S**. Authors using the interactive strategy then enter the outer loop, where their reading of **A** prompts a new idea in **S** which they draw down and re-enter the inner loop. Authors using the outline-first strategy do not use the outer loop since they are drawing down a complete story draft from **S**. Our data suggests that additional schemas comprising linguistic knowledge (rhetorical features as well as lexical and semantic) feed into this dynamic process.

Let us consider the evidence for this model. Examples of stories composed using the outline-first strategy are the *Rain Dance* and *The Rescue*. The *Rain Dance* author indicated she started thinking about what to do with character movements and suddenly the complete idea of a rain dance arrived in mind. She then entered the inner feedback loop with many revision cycles, to obtain the dance she wanted. The author of *The Rescue* indicated he started with appealing images, an injured Robin and a blazing fire and this led to the complete rescue idea. He indicated that he then ‘saw the movements’ in mind, suggesting he was simulating the story before coding it. Then he entered the inner loop and stayed there until his story was complete. Interestingly, both authors referred to the affordances of the image and movement modes as suggestive to establishing their story drafts.

Let us consider an example of an interactive strategy used by Emily (pseudonym) with whom we worked in laboratory conditions intensively over three sessions of two hours, together with her mother. Emily enjoys writing stories, and told of her interest in developing characters and relationships in her stories. We asked Emily to try to forget any recent compositions or reading, and to code a story using the coding she had learned, but to express herself as she wanted. We observed Emily as she was coding, recorded her comments but did not interrupt or ask questions so as not to disturb her activity. Her story is transcribed below, italics indicate embellishments.

Jig and Pip are paddling by the lake. They walk out the lake to near the mushroom, and rest for a bit.

Pip says 'What shall we do now?' Jig says 'Let's fly up to the saucer' so we can explore it.

He flies to the saucer and rests a while. *But Pip wants to see the scarecrow instead because she doesn't like flying.*

It makes her tired. She walks over to the scarecrow.

Jig spins a bit too fast by the saucer *because he is impressed because of the flashing lights and colours.*

Pip then flies up to see Jig *because she forgets about it making her tired.*

Pip spins *because she is excited too.* They look around the saucer.

Jig wants to go and see the scarecrow. They fly to the scarecrow and stay there for a bit.

Then Pip says 'What shall we do now?'

They decide to walk back into the lake and relax *as it is such a hot day.*

The end.

We discussed this transcription with Emily (age 8) to try to understand her composition strategy. Our comments are shown [bracketed]. She indicated the lake image was appealing and placed the actors there to infer paddling. She coded movement to the appealing mushroom image and then she felt the need for choice [expressed through verbal embellishment with reason]. She wanted to develop Jig and Pip as different characters so she made them move initially to different places [and provided reason]. She then resolves the 'character conflict' and gets them to move to the saucer, and then to the scarecrow [again with reason]. She felt the need for choice again [verbal embellishment] and she makes the decision to return the actors to their starting place [with reason]. Emily in this discussion suggested strongly that she had used an interactive strategy, choosing her next story idea based on the animation developed so far, and was keen to explain that she felt the need to express cause or reason.

This story exhibits characteristics we have found in most of the interactive strategy compositions: (i) the existence of temporal or causal links (of short span) and (ii) the existence of choice but often of an almost random nature. While we were able to assign the planned stories into types (e.g., 'rescue', 'defeat', 'argument') the emergent stories could not be easily classified. Indeed classroom teachers referred to these stories as 'clunky' in need of further refinement.

Formatted: Heading 2

Story Initiation

Finally we looked at how the authors initiated their stories. Analysis of the corpus shows that 14/20 of the stories used the affordances either of the images (scenery or characters) or of character actions, or both in framing the story through code. Examples of openings based on image affordances are; '*There was the most beautiful flower ever through all of land', 'Once upon a time … he was about to jump into the lake', ' Drax … was standing … close to the fire', 'It was a lovely sunny day'*. Examples based on action affordances are; *'Grog and Pip are playing …'* (5), *'Grog was having a lovely day, jumping up and down …'*. Combined image and action affordances include, *'[they] all went to the park, despite it being cloudy'*.

In summary, these authors seem at home with 'new literacies' able to critically select the media affordances to initiate and progress their composition, and engaged in a process of *design*. Working in the inner loop it was clear that the children were carrying out 'what-if' simulations, like those proposed by Wohlwend (2015). Within the usual classroom setting, children were keen to view other's stories and some collaborated sharing ideas, solutions to problems and some even shared short 'snippets' of code. We also found that they moved fluently between the coding screen and the animation screen and did not seem to require excessive cognitive effort in doing this, see Jewitt and Adamson (2003)**.**

Findings and Implications

We were initially interested in two questions, first how children could express a story using the medium of coding, and second the strategies used in composition. We have found that children were using code in ways we did not expect, especially through inference and implication. We have found two clear strategies of composition, outline-first and interactive which seem to produce different flavours of narrative text. More research needs to be done here, especially looking at the age and maturity of authors.

We have demonstrated that young children can master some of the syntax and semantics of a high-level professional programming language in the context of writing stories. We feel this has implications for how coding is taught in the classroom, SWC makes coding less abstract and brings it alive. So here is evidence of Burnett's suggestion that new literacies can support creative coding (Burnett, 2016). In addition we suggest that coding can develop children's multimodal authoring skills, though we do suggest there is evidence of the inverse; coding can help composition through encouraging themchildren to be critical of media affordances, to through activities which encourage design, experimentation and also to play. Therefore we see SWC as having cross-curriculum implications for classroom practice.

We have provided additional support for multimodal research suggesting how children have used the affordances of print (**C** and **S'**) together with static and moving images (**A**) cohesively to produce stories but where the print-text has a central role.

There are some limitations to this research. We have made no attempt to understand our initial observation that children spontaneously told stories while coding. There should be an investigation of whether such an engine can support the telling of classic tales, poetry, drama productions or silent movies. Also the research was not conducted within the curriculum; we are currently addressing this by teaching a Yr3&4 class of 25 children as part of their curriculum for a whole term. While teachers were very enthusiastic about how the children were focussing on coding stories, some were less impressed by the "emergent" stories obtained, when written down. Criticism of these texts included (i) lack of character development, (ii) lack of required grammar and

Formatted: Normal, Justified

choice of 'interesting' words. We have responded to (i) where the latest engine release allows characters to display emotions, and to write speech and thoughts to the screen. Concerning (ii) we feel that the teachers were perhaps responding uncritically, given the constraints of having to provide assessment evidence, such as the use of 'big' words and fronted adverbials. It may be that these "emergent" stories are indeed constrained by the limited actions available in the engine. They may only exist when read together while viewing the animation, in which case they are true multimodal stories. Further research should investigate this. Finally our process model needs further investigation.

In conclusion we hope to have provided evidence that computer coding can be viewed as a form of literacy and therefore taught in the classroom in a creative and not abstract sense, and therefore SWC may provide support and challenge across the curriculum. Hopefully we may inspire others to welcome computer coding, as a creative activity, into the fold of new literacies and even join us in our quest. To this end we are happy to provide the engine, selected stories and instructional materials upon request. The only requirement is a PC (not a Mac); no specialist software needs to be installed. Please contact the authors directly.

References: see the journal's instructions for authors for details on style

Alexander, R.J. (Ed.), (2010). *Children, their world, their education: final report and recommendations of the Cambridge Primary Review.* London: Routledge.

Bereiter, C., & Scardamalia, M. (1987). *The Psychology of Written Composition*. London: Lawrence Erlbaum.

Burnett, C., Davies, J., Merchant, G., & Rowsell, J. (Eds.) (2014). *New Literacies around the Globe: Policy and Pedagogy.* Abingdon, Oxon: Routledge.

Burnett, C. (2016). *The Digital Age and its Implications for Learning and Teaching in the Primary School.* York: Cambridge Primary Review Trust.

Department for Education (DfE). (2013). *National curriculum in England: computing programmes of study*. London: Department for Education.

Elbow, P. (1981). *Writing with Power*. New York: Oxford University Press.

Elbow, P. (1998). *Writing without Teachers*. Second Edition. London: Oxford University Press.

Galbraith, D. (1999). Writing as a knowledge-constituting process. In M. Torrance & D. Galbraith (Eds.), *Knowing what to write* (pp139-160). Dordrecht, NL: Amsterdam University Press.

Halliday, M.A.K. (2004). *Halliday's Introduction to Functional Grammar*. Oxford: Routledge.

Hayes, J.R., & Flowers, L. (1980). Identifying the organization of writing processes. In L. Gregg & E. Steinberg (Eds.), *Cognitive processes in writing: An interdisciplinary approach* (pp 3-30). Mahwah, NJ: Lawrence Erlbaum.

Hayes, J.R. (1996). A new framework for understanding cognition and affect in writing. In C.M. Levy & S. Ransdell (Eds.), *The science of writing: Theories, methods, individual differences, and applications* (pp. 1-27). Mahwah, NJ: Lawrence Erlbaum.

Hayes, J.R. (2006). New Directions in Writing Theory. In C.A. MacArthur, S. Graham & J. Fitzgerald (Eds.), *Handbook of Writing Research* (pp 28-40). New York: Guilford Press.

Husbye, N.E., Buchholz, B.A., Coggin, L.S, Wessel-Powell, C., & Wohlwend, K.E. (2012). Critical Lessons and Playful Literacies: Digital media in PK-2 classrooms. *Language Arts*, *90(2), 82-92.* Retrieved from http://www.jstor.org/stable/41804380

Jewitt, C., & Adamson, R. (2003). The Multimodal Construction of Rule in Computer Programming Applications. *Education, Communication and Information*, *3(3*), 361-382. Retrieved from http://dx.doi.org/10.1080/1463631032000149683

Kress, G., & Van Leeuwen, T. (2006). *Reading Images: The Grammar of Visual Design* (2nd ed.). Oxford: Routledge.

Marsh, J. (2010). Young children's play in online virtual worlds. *Journal of Early Childhood Research,* 8(2), 23-39. DOI: 10.1177/1476718X09345406

Marsh, J., Plowman, L., Yamada-Rice, D., Bishop, J.C., Lahmar, J., Scott, F., & Winter, P. (2015). *Exploring Play and Creativity in Pre-Schoolers' Use pf Apps: Final Project Report.* Retrieved from http://www.techandplay.org/reports/TAP_Final_Report.pdf

Merchant, G. (2007). Writing the future in the digital age. *Literacy, Vol. 47(8),* 118-128. **DOI:** 10.1111/j.1467-9345.2007.00469.x

Resnick, M. (2002). Rethinking Learning in the Digital Age. In G. Kirkman (Ed.), *The Global Information Technology Report: Readiness for the Networked World* (pp 32-37). Oxford: Oxford University Press.

Resnick, M. (2013). Learn to Code, Code to Learn. Retrieved from https://www.edsurge.com/news/2013-05-08-learn-to-code-code-to-learn

Sefton-Green, J., Marsh, J., Erstad, O., & FLEWITT, R. (2016). *Establishing a Research Agenda for the Digital Literacy Practices of Young Children: a White Paper for COST Action IS1410.* Retrieved from http://www.lse.ac.uk/media@lse/research/pdf/COST-2016.pdf

Unsworth, L., (Ed.) (2008). *Multimodal Semiotics Functional Analysis in Contexts of Education*. New York: Continuum.

Walsh, M. (2009). *Pedagogic Potentials of Multimodal Literacy.* In L.T.W. Hin & R. Subramaniam (Eds.). *Handbook of Research on New Media Literacy at the K-12 Level: Issues and Challenges* (pp 32-47). Hershey, NY: IGI Global.

Wohlwend, K.E., (2015). One Screen, Many Fingers: Young Children's Collaborative Literacy Play With Digital Puppetry Apps and Touch Screen Technologies. *Theory Into Practice, 54(2),* 154 – 162. Retrieved from http://dx.doi.org/10.1080/00405841.2015.1010837

Table 1. Type your title here.

Figure 1. Type your title here. Obtain permission and include the acknowledgement required by the copyright holder if a figure is being reproduced from another source.

1	*“Flup went to Grog”*	**flup.walkto(grog);** **pip.rest();**
2	[pause] *“because they are friends”*	
3	*“Then Pip went over to Grog”*	**pip.hopto(grog);**
4	*“Then went to the egg*	**pip.runto(myegg);**
5	[pause] *and broke it”*	

Table 1. An example of a spontaneous story told by a child while coding. Story clauses are shown on the left with associated code on the right.

Moves To	**Moves At**	**Pause**	**Existential**	**Mental Perception**	**Relational Intensive***	**Acts On***

walkto();
runto();
flyto();
hoptto();
	jump();
	spin();
		rest();	add();	sees();
				isnear();
					is();
					feels();
						pickup();
						putdown();

participant.process(circumstance) **grog.flyto(scarecrow);**						

Table 2. Code statements available to the writers organised according to the *processes* of the ideational metafunction. Brackets indicate where parameters (*circumstances*) can be inserted. Statements indicated * were not available to all writers.

Story 17 *The Rain Dance*		
1	*Pip and Grog were much too hot.*	
2	*But luckily they both knew a rain dance*	
3	*to help them cool down.*	

4	*Grog and Pip did their dance.*	grog.jump(5);
		grog.spin(2);
		grog.rest();
		grog.walkto(30);
		grog.runto(40);
		grog.hopto(5);
		
		pip.jump(5);
		pip.spin(2);
		pip.rest();
		pip.walkto(50);
		pip.runto(40);
		pip.hopto(80);

5	*And as they had wanted*	

6	*lots of clouds appeared.*	add(cloud,30,50);
		add(cloud,55,45);
		add(cloud,35,40);
		add(cloud,75,45);

7	*Pip and Grog were much happier*	
8	*when there was clouds to keep them cool.*	

Table 3. *The Rain Dance.* Story clauses are shown on the left with associated blocks of code on the right. Lines without associated code are “embellishments”.

	Story 13 *The Rescue*	**Code**	**SysNet**
1	*Grog was having a lovely day,*		implied
2	*he was jumping around on the ground*	D	inferred
3	*Pip was on the hillside*	C	derived
4	*enjoying a mountain view*		embellishment
5	*when he spotted something in the distance*		implied
6	*an injured Robin up a tree.*	A	derived
7	*He also noticed a strong blaze surrounding the tree*	B	inferred
8	*He signals to Grog to fly up*	E	inferred
9	*and help get the Robin and save him.*		embellishment
10	*As grog hears he immediately begins to put on a brave face.*		embellishment
11	*He flies up to the tree*	F	inferred
12	*and allows the robin to get onto his back*		implied
13	*"You'll be safe with me" exclaims Grog.*		embellishment
14	*Grog flies off down to the ground*	G	direct
15	*and lets the robin get off.*	H	inferred
16	*"Thank you so much, I surely would have burned to a crisp if you weren't there to save the day" utters a relieved robin.*		embellishment
17	*"I wouldn't have known to help if Pip wasn't*		embellishment

A	add(bigtree,70,10);
A	add(myrobin,63,37);
B	add(fire,60,10);
B	add(fire,73,9);
B	add(fire,68,10);
B	add(fire,64,10);
C	add(pip,20,40);
D	add(grog,35,10);
D	grog.hopto(60);
D	grog.runto(10);
D	grog.rest();
C	pip.rest();
C	pip.rest();
E	pip.spin(4000000);
F	grog.hopto(45);
F	grog.flyto(myrobin);
F	myrobin.rest(10);

Table 4.*The Rescue*. Narrative text is shown on the left with keys (A to H) showing mapping onto the code on the right. Also shown is the system used to map meaning for each clause or sentence, from the sysnet (Figure 3).

H	grog.rest();

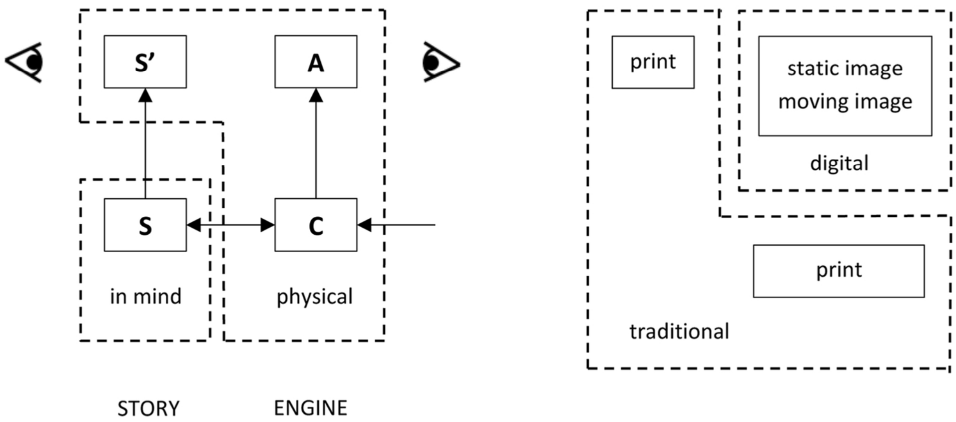

Figure 1. Left illustrates the structure and user interaction with the SWC process. The engine components are on the right; C – code text input, A – animation. The story S and its narrative text S' are on the left. Eyes show channels through which the story is read. Dotted regions show which components are 'in mind' and in physical space. Right maps the physical components onto modes and dotted regions show which modes can be thought of as 'traditional' or 'digital'

69x32mm (600 x 600 DPI)

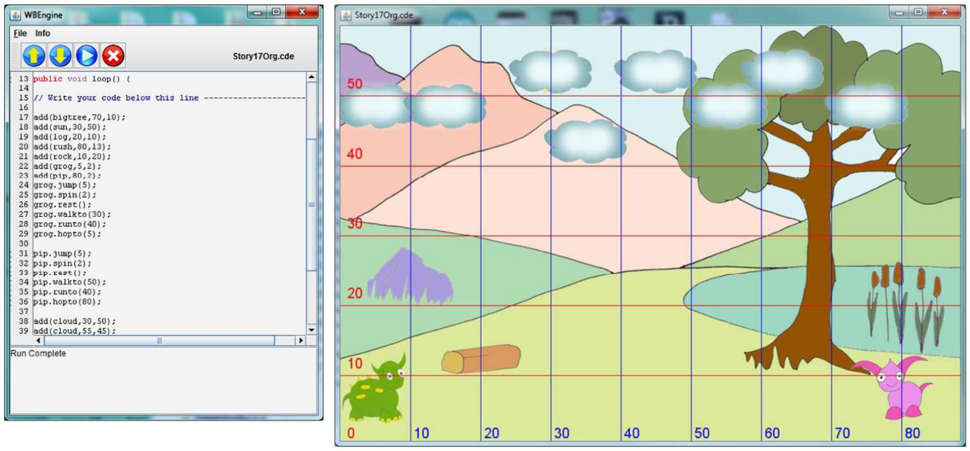

Figure 2. Appearance of the SWC engine on the computer screen. The text box for code entry is shown on the left and the animation canvas on the right. The grid overlay with Cartesian coordinates to help children locate entities is shown enabled.

261x125mm (300 x 300 DPI)

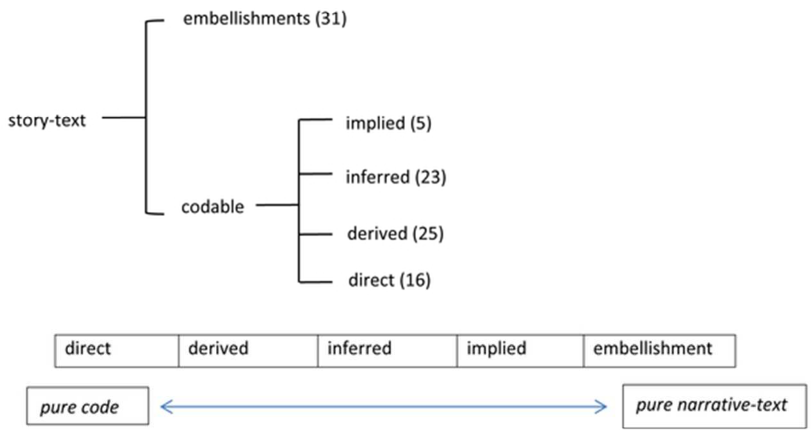

Figure 3. The system network derived from analysis of our corpus. Story meaning is expressed as non-codable text ('embellishments') or through codable text through four additional systems. Parentheses show the number of examples of each system found in the corpus expressed as a percentage of the total number of clauses in the corpus.

30x19mm (600 x 600 DPI)

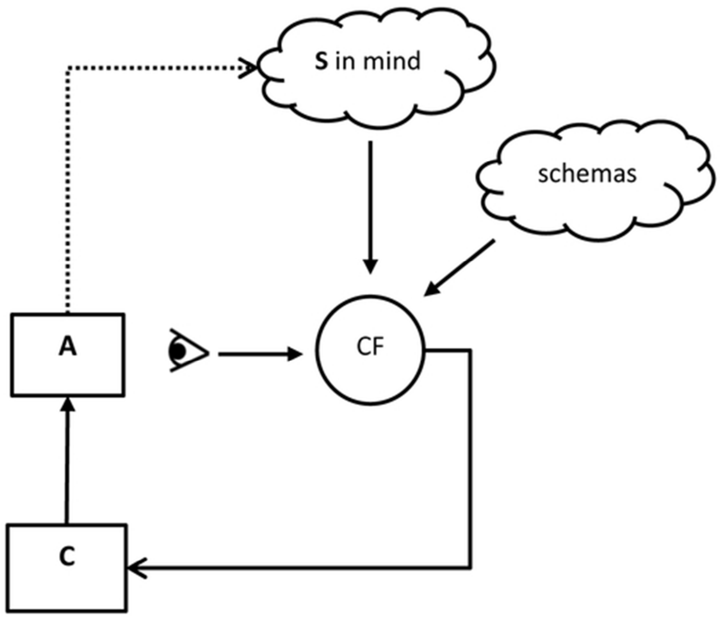

Figure 4. Proposed model of the SWC process. Solid arrows show an inner loop of activity where children write code C, read the animation A produced, make a comparison (CF) with the idea S they have in mind and perhaps revise the code. Some children enter the outer loop (dotted arrow) where A prompts a new story idea S which is then coded. Others start with a complete draft story in mind S and do not use the outer loop. We propose they also draw down linguistic schemas from mind.

30x27mm (600 x 600 DPI)