A Case-Based System to Aid Cognition and Meta-Cognition in a Design-Based Learning Environment

Ganesh P. Bhat and Janet L. Kolodner

College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332
ganeshbhat@gatech.edu, jlk@cc.gatech.edu

Abstract
Design-based learning (DBL) has many affordances for promoting deep and lasting learning of both content and complex skills. However, careful orchestration and scaffolding are usually needed to achieve its full potential. In this paper, we describe our efforts at implementing a software suite to meet the cognitive and meta-cognitive needs of learners engaged in DBL. In Study 1, our software suite gave learners the opportunity to design in simulation, to run experiments to learn the effects of variables, and it scaffolded science explanation construction. Through our analysis of study 1 we identified both cognitive and metacognitive needs that the software did not provide for. To meet these additional requirements, we added an interactive science resource and a case library to the software to provide multi-representational content material, to facilitate exploration, and to invite metacognitive reflection needed to do well at learning through design. Learners recognized what they did not understand, took initiative to explore those science concepts, and applied them in novel ways. We present here our analysis of the kinds of metacognitive help learners need to productively learn from design activities and some ways of providing that help. Our conclusion is that cognitive aid without related metacognitive aid is insufficient in a DBL environment.

Introduction
In design-based learning [1], learners learn content and skills in the context of designing some working artifact. The design challenge provides a context for becoming curious and reason to engage. The experience itself provides a venue for practicing new skills and applying newly learned concepts. In addition, the process of iteratively working towards achieving a design challenge promotes deliberative reflective practice – identifying what one does not understand well, seeking to come to a better understanding, and applying one’s new understanding to achieving the goal. Achieving design challenges requires understanding and application of content, carrying out complex reasoning, and significant collaboration and communication, design-based learning (DBL) has powerful affordances for promoting deep and lasting learning of both content and complex skills.

But getting the full set of affordances out of DBL requires careful planning and scaffolding. What makes DBL potentially very powerful for promoting learning is what also makes it difficult to enact productively, whether in a classroom or an informal learning environment: the combination of learning well while designing and successfully achieving a design challenge requires that the learner carefully regulate a whole variety of cognitive goals. In DBL, cognitive and metacognitive reasoning are complexly intertwined with each other. Cognitive strategies are used to help an individual achieve a particular goal (e.g., choosing a component) while metacognitive reasoning is required to ensure that the goal is reached (e.g., monitoring one’s progress and deciding what to pursue next).

In DBL, metacognition comes up in several ways. A learner may question herself about the concepts required to successfully complete the design. Her cognitive goal is to be able to design. If she finds that she cannot answer her own questions, or that she does not understand the material discussed, she must then determine what needs to be done to ensure that she meets the cognitive goals that must be achieved. She may decide to read something or ask an expert to answer questions she had generated. Once she understands something, she must decide if she needs to further understand something else or go back to designing. A metacognitive strategy of self-questioning is needed to ensure that the cognitive goals of designing are met. Or, a learner may use meta-cognitive knowledge in moving through the design process. For example, planning how to approach the problem of removing spin from a flying saucer: "I know that I (person variable) have difficulty with..."
spin (task variable), so I will design and build the flying saucer first and save the problems with spin for last (strategy variable)."

Metacognitive and cognitive reasoning may even overlap. For example, one may use a self-questioning strategy while reading as a means of obtaining knowledge (cognitive), or as a way of monitoring what one has designed (metacognitive). Because cognitive and metacognitive reasoning are closely intertwined and dependent upon each other, any attempt to examine one without acknowledging the other, we believe, would not be adequate.

We subscribe to the normative definition of metacognition that says that metacognitive strategies [2] are sequential processes that one uses to control cognitive activities, and to ensure that cognitive goals are being met. These processes help to regulate and oversee learning, and consist of planning and monitoring cognitive activities, as well as checking the outcomes of those activities. However, as we will show, we do not believe that one should design cognitive and metacognitive aid for learners separately from each other. Rather because those needs are so intertwined in design-based learning, aid to learners must be designed to address learners’ cognitive and metacognitive needs in conjunction with each other.

Research Goals
To harness the full power of Design Based Learning, we need to understand what kinds of scaffolds are needed so that learners can effectively plan next steps and gauge their progress, and when metacognitive aid is best deployed. We focus our research on the following issues:

- Determine how different scaffolding methods invoke self-regulatory processes that facilitate students' learning of challenging science topics.
- Study how different scaffolding conditions influence learners' ability to regulate their learning of complex and challenging science topics.
- Distribute scaffolding roles between the software and external regulating agents (i.e., human tutors and peers).

Because we take a case-based approach, we also seek to determine how a case library can best be designed to address the needs of learners as they are designing.

Approach
We do our work in the context of helping middle school kids (ages 12 to 14; grades 6 to 8) learn about forces and motion and elementary fluid dynamics in the context of designing hovercrafts. In particular, we build on two approaches to promoting learning through design: Papert’s Constructionism [7] and Kolodner’s Learning by Design [6, 8, 9]. Constructionism, developed from a Piagetian perspective, points out the affordances of learning through designing something personally meaningful and the important roles that an audience plays in promoting learning through designing. Its principles have been used to design a large variety of software, hardware, and communication infrastructures that successfully support learning in informal learning environments.

Learning by Design (LBD) focuses on the sequencing and orchestration needed to make design-based learning work productively in middle-school classrooms. Learning by Design was designed based on the cognitive model implied by Case-Based Reasoning [3, 4]. CBR defines learning as extending one’s knowledge by interpreting new experiences and incorporating these experiences into memory. A case-based reasoner learns by interpreting, collecting, and reusing its experiences and the lessons learned from them. Important to such learning is exploration in pursuit of personal goals, interpretation of one’s experiences with respect to achievement of those goals, and repeated opportunities to reuse one’s experiences and continue to interpret them over time to draw out lessons learned.

Both approaches tell us that for learning from design to be effective, it is critical that learners have the opportunity to build and test real artifacts and that design activities be iterative, giving learners the chance to redesign based on feedback – both the feedback that comes in testing their designs and the reaction and advice of their audience. In Learning by Design curriculum units, students aim towards achieving appealing design challenges (e.g., design a vehicle and its propulsion system that can navigate several hills). The sequencing promotes systematically uncovering and working towards sub goals as they move towards their design goal and systematically interpreting their experiences with respect to those sub goals as they go along. Doing and reflection are interwoven with each other. Several “tricks” are used in LBD to promote reflection. Among them is dividing up investigative work among groups in a class so that learners are motivated to attain the expertise they need to advise their peers, to make presentations that their peers will understand, and to listen to their peers to learn from them. Two other “tricks” are helping learners recognize early on that the advice they can get from peers helps them achieve their design goals more easily and in more satisfying ways and having them experience the satisfaction of others giving them credit for ideas they have contributed. These tricks, in turn, encourage learners to reflect deeply on experiences they are having so that their presentations will affect their peers’ understanding and designs.

We have created an LBD-like “curriculum” called Hovering Around. Learners work towards achieving a variety of successively more complex hovercraft challenges. Each design challenge promotes curiosity about a subset of hovercraft science concepts. For example, curiosity about how to get a craft to lift higher promotes investigation of issues related to Newton’s Laws of motion. Learners divide up the work of investigating and report to each other as in LBD. They build physical
hovercrafts and also explore hovercraft design and hovercraft science using software that allows the same kinds of exploration and investigation they might do in the physical world but requires less time, allowing some things that are impossible in the world, and provides access to phenomena that are invisible in the world (e.g., forces).

In the Hovering Around curriculum, learners begin by discussing what hovering is and what a hovercraft is. They see a short video about hovercrafts. Then they progress through a series of small challenges, each focused on designing and building a small model hovercraft: a balloon craft, then one that can lift but does not go forward, then a forward-moving craft. They address a variety of issues as they were doing this — making their crafts stable, getting the spin out, making them go higher, making them travel over a variety of surfaces, and so on. Each issue promotes curiosity about a set of science concepts, e.g., net force, lift, power, equal and opposite forces, and air flow. Sequencing has learners moving from large group work to small group work and back again. When meeting as a large group, they work around a Project Board, keeping track of what they think they know, what they need to investigate, what they are learning, their evidence for those things they are learning, and how to use what they are learning to achieve their goals. They decide as a whole group what the critical questions are for designing each small craft and then work in small groups to run experiments to answer those questions. They report to each other with the help of a poster session about their results.

We began our studies knowing that learners in a DBL environment need help with identifying next sub-goals to work on and explaining phenomena they are experiencing, and that deeply learning the science and learning to explain require repeated deliberative practice, i.e., practice that they reflect on to identify what they are doing well and what they need to do or understand better. Our finding from LBD was that the better teachers are at providing that kind of help, the more deeply learners learn the science and the better they are able to form scientific explanations [5]. We thus sought in this project to design software that could play roles to complement teacher abilities.

**Study 1**

Our first approach to our hovercraft curriculum was to design and integrate into an LBD-type curriculum a software package called Jackets’ Garage. Jackets’ Garage has support for designing and simulating performance of 3 types of hovercrafts — a balloon hovercraft, a flying saucer (a craft that lifts but does not move forward), and a forward-moving craft. The software was designed primarily to allow learners to do more exploring in less time than they can using physical materials alone. It also was designed to guide them through running experiments to investigate phenomena that affect hovercraft performance, to help them see physical phenomena, and to construct proper scientific explanations. Our idea in designing the software was that learners would begin each hovercraft design challenge by constructing a hovercraft out of physical materials. These never work exactly as they want, so this experience would raise questions. They would then use the software to explore and investigate to answer the questions that were raised. The software consists of 3 sections; Garage, Experiment Lab, and Racetrack.

The Garage is a modeling area where learners build virtual hovercrafts, manipulate their variables and simulate their designs. Learners can build hovercrafts that are consistent with the real-world physical parts they have available, or they can build imaginary ones, e.g., with batteries that have no mass. This allows exploration to answer “what if” questions. A design decision chart provided in the garage is designed to help learners both cognitively and metacognitively. The chart provides space for noting design decisions and space for recording the purpose (goal) of this design feature and the evidence (from investigations) and science knowledge they have that justify the decision.

![Fig. 1 Experiment Lab](image)

The Experiment Lab [Fig. 1] provides structure for designing and running experiments to identify the affects of important variables, e.g., mass of the craft, power from the engine, on a hovercraft’s performance. Metacognitively, this tool is laid out to help learners know what are the possible next steps, to walk them through steps involved in experimentation, and to point them to a need to scientifically understand experiment results and extract rules of thumb that could be used for designing. The learners are taken through the steps of a formal experiment where they chose the dependent and independent variables, manipulate the variables, generate a hypothesis, make observations and use these observations to construct a conclusion.

The Racetrack allows learners to race their own crafts with other crafts built by them or other learners. This component also encourages prediction and explanation.

Jackets’ Garage was to be used when learners needed answers to science questions to successfully redesign their physical crafts, or when they wanted to try out a design idea before building it. Within each section, metacognitive
help is provided to learners through the sequencing of pages (in the journal), the setup of tables (in the garage), and the questions learners were required to answer before moving on (in the journal and the racetrack).

**Study 1 Result:** We rolled out the software in an after-school program in 2007-2008 and analyzed the participants’ learning and software use. We had a small group of mostly 6th graders who were enrolled in aftercare at the school. We found successes and failures.

- Learners successfully learned some hovercraft science.
- Learners used the software try out both technical and science ideas.
- The software was easy to use, but successful explanation required a lot of help from the facilitators.
- Learners were frustrated that some issues they wanted to explore (e.g., preventing their hovercraft from spinning) could not be explored in the software.

However, learners had more trouble than we expected at articulating explanations. Learners needed more in the way of examples, explanations presented in a variety of ways, and help with applying science to explain hovercraft phenomena. The software presented definitions but not explanations of science concepts. In addition, there was much hovercraft science that was too complex for us to build into a simulation but that learners wanted to explore.

**Moving forward from Study 1**

On the other hand, we realized early on during Study 1 that the software encouraged learners to wonder about the science but that the science content was missing from the software tool. And we realized that the learners needed reminders about the science they had already learned. Most importantly, we recognized that if we continued having learners use the software the way we had intended and kept track of the help we (the facilitators) were providing, we had the chance to identify what additional help learners needed and how that help might be provided in software.

This analysis pointed out many interesting things. First, one of the most prevalent strategies we used as facilitators was providing examples familiar to the learners to make the science clear. Sometimes we sketched those examples and spent a lot of time walking through them. Sometimes we waved our hands through examples, assuming they were familiar to the learners but not actually sketching examples. We also built and brought to the learners variations that they could not examine in the software (e.g., different types of skirts and their effects). In all of this, we were helping learners activate prior knowledge and handle task demands. We noticed, too, that the hovercraft designs learners were creating were quite close to the designs we discussed with them.

We wondered if there was a way to help learners be more inventive in their designs. On this issue, we thought providing a case library would help. We also wondered how we could help our software provide more of the kinds of help we were providing to learners. Our initial goal, after all, had been to complement what teachers could provide, and many teachers, we knew, would not be able to provide the science and technology knowledge some of our facilitators had. Even when teachers can be resources for such knowledge, no teacher can be available to every learner in a class when the learner needs them. For this, we thought a library of examples learners are already familiar with, indexed to the concepts they illustrate, would be useful. The examples, we decided, not only needed to include illustrations, be we thought the learners needed to be able manipulate the relevant systems of each, e.g., attach second propellers to a helicopter in several different orientations to see how spin is affected. While it was too difficult to simulate every variable that might affect hovercraft performance in *Jackets’ Garage*, we thought that we could be successful at separating out variables and providing learners the opportunity to manipulate those variables and wondered to what extent learners would then be able to integrate what they learned into their hovercraft designs.

**The New Software: Creativity Clubhouse**

We designed a piece of software called *Creativity Clubhouse* to address these needs. There are two parts to *Creativity Clubhouse*:

- A Resource Library that consists of science and engineering articles.
- A Case Library of real examples of hovercrafts. The cases highlight the issues designers faced while designing and debugging their hovercrafts.

![Fig. 2 An article on hovercraft skirts.](image)

The resource library was designed with three purposes:

- To provide science and technology content learners need to understand the results of experiments.
- To provide suggestions about what notes they might need.
To provide guidance as they explored science and technology content, so as to encourage deep understanding.

The resource library has articles about the parts of a hovercraft, the science behind each of those parts, and the science of how a hovercraft works. We made lists of the questions learners had about each of the concepts and objects in the resource library and organized articles around those questions. Each page has five parts: text that answers the question, an interactive demonstration, a real-world example, in the form of a video or an interactive simulation, a space for note taking, and pointers to other related articles. The articles were interactive and used multiple representations to describe relevant concepts [Fig. 2]. Each article provides help to learners in constructing mental models (cognitive activity), and each is organized around questions learners might have in their minds, suggests explorations they might engage in, and suggests other articles they might find relevant (metacognitive help).

The case library was designed to serve four purposes:

- To help learners have an imagination about what they might design.
- To give learners ideas about issues they need to address.
- To provide pointers to relevant science and technology articles.
- To provide learners suggestions about what they might want to take notes about.

Cases are each divided into four sections.

- The overview section introducing the learner to the case and presenting the design challenge the people involved were trying to achieve.
- The issues encountered section presents the key issues that the designers ran into while working towards their design solution. It lists the issues and allows the learner to focus on each of these independently. Each of these issues have supporting videos that present the issue in detail and how the designers dealt with it.
- The parts section discusses the physical parts that went into the final design.
- The results section summarizes the final designs and presents the performance of the craft.

The cases in the current case library are pulled from popular TV shows. In each, some “cool” expert discuss what he or she was trying to accomplish for a competition, the issues that arose, and how they dealt with each.

**Study 2**

Our goal in Study 2 was to learn to what extent our newly developed tools provided the cognitive and metacognitive support we knew learners needed and to identify if there was other support they needed in addition. We had nine learners join us for eleven 3-hour sessions. The learners were home schooled, and their parents were looking for supplementary science experiences for them. We provided them with laptops and all the materials required to build the physical hovercraft. Data was collected in the form of video recordings; screen captures and field notes written by the facilitators.

The curriculum was as in the first study, with two exceptions. In the first study, learners worked on a succession of harder and harder hovercraft design challenges, culminating in their design of a craft they could ride on. This time we told the learners up front that the main goal was to build a hovercraft they could ride on and that the other crafts they would build would help them learn what they needed to learn to design and build the large craft. Also, in the second study learners had Creativity Clubhouse available to them, which afforded some additional taking on of expertise that learners did not do in Study 1.

**Results**

Many of the drawbacks of having just Jackets’ Garage were overcome with the help of Creativity Clubhouse. The facilitators could now act more as guides and less as resources, as the articles in Creativity Clubhouse took over the role of providing most of content help and also much of the metacognitive help facilitators provided in Study 1. The learners were now able to themselves access the resources they needed to make explanations in Jackets’ Garage and also make informed decisions about how to improve their designs based on their experiences with the cases. That the software was doing much of what we intended could be seen through several indicators:

- Learners’ construction of robust science conceptions
- Learners’ more thorough understanding of their designs
- Learners’ navigating from cases to science articles when they lacked understanding
- Learners’ navigating to science articles to make explanations in Jackets’ Garage
Gains in declarative knowledge from pretest to posttest
• Observed shifts in learners’ mental models
• Learners’ more novel and creative designs.

We also found that some things were hard for the learners without a facilitator guiding them. The learners initially found that videos in the case library distracted them from efficiently utilizing the full software but once they were provided with the necessary sub-goals, we noticed the learners utilizing all parts of the software. Note taking did not happen until a facilitator stressed its importance, and few learners made a significant effort to take notes except when they knew they had to be expert for the class on a particular tough concept. Also, most of the learners did not see the need to explore unless they were provided with smaller goals. We saw that our approach with Study 2 had a lot more of learning when compared with study 1. We believe that the better results were due to the introduction of the Creativity Clubhouse as a resource. Jackets’ Garage worked well with the small individual goals but did not really answer the big questions the learners had when it came to building a hovercraft they could ride on. Creativity Clubhouse provided the learners with all the resources needed to answer those big questions. The software along with the curriculum allowed the learners to closely monitor their progress (with the help of design charts and Project Board) and return to it to evaluate ones thinking every time their experiments or designs failed. We were also delighted to see how Jackets’ Garage and Creativity Clubhouse complimented each other.

Conclusion
It seems significant to us that there has never been any area of our design where we felt able to simply design for cognitive needs and not take the metacognitive needs of learners into account at the same time. Just as people who are studying cognition are becoming more and more convinced that you cannot actually teach metacognition, we are becoming more and more convinced that you cannot scaffold either cognitive or metacognitive reasoning by itself, but that the best software-realized scaffolding is designed to provide both types of help together and that only by inviting metacognition as it is needed to achieve cognitive goals will we be able to help youngsters become more able metacognitive reasoners.

Acknowledgements
This research was partially supported by a grant from NSF. We thank Chandan Dasgupta, Lee Chiem, Jim Perkins and other participants in the project.

References