

# Developments in Interactive Visualization and Physics Simulation with Cinderella<sup>1</sup>

University of Potsdam, Institute of Mathematics, Chair of Mathematics Education,  
Ulrich Kortenkamp

## 1 Introduction

In this article I will describe some future directions of the Interactive Geometry System Cinderella, summarizing the presentations at the RIMS workshops in 2014 and 2016. Cinderella [10] is a DGS that has been published in Japan [11] in 2003. Instead of describing the system in full depth I refer to a presentation given at a RIMS workshop in the same year. In [1] we describe the current state of Cinderella, some advanced learning scenarios, and we give an outlook for the future of such interactive teaching tools. In this article I will recall some of the aspects presented there and how these have been implemented and used — or not — during the last decade. I will also describe some of the current challenges for DGS, in particular in the areas of 3D input and output and how technology development will enable us to blend the physical world with abstract microworlds realized in a DGS. Furthermore, I will point out how technological development made it possible to still present Cinderella content on the web, in spite of the death of Java on the client side.

## 2 A decade of DGS

Since the introduction of DGS in the end of the 1980's more than 30 years have passed, and some teaching scenarios that have been described in 1990's have become commonplace in the classroom. In 2001 the DGS GeoGebra was created by Markus Hohenwarter and since then it became popular, in particular because it has later been published as Open Source Software, as opposed to the DGS Geometers' Sketchpad and Cabri G?om?tre. Currently, GeoGebra is the predominant software available and many people are convinced that it is the first, the only, and the best DGS that has been and is available. While this is not true in its generality, it can be acknowledged that GeoGebra was able to make DGS a tool that is being used regularly in many classrooms.

---

<sup>1</sup>The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 610467 — project “M C Squared”. This publication reflects only the author's views and Union is not liable for any use that may be made of the information contained therein. The author has been supported by the Japan Society for the Promotion of Science under a Short Term Fellowship for Research in Japan.

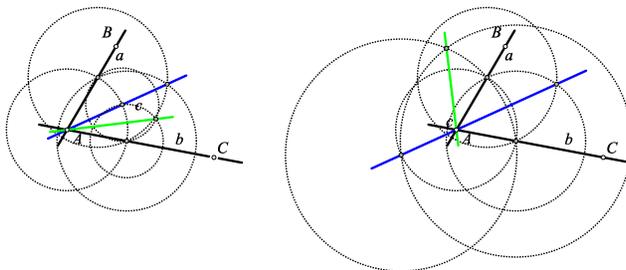


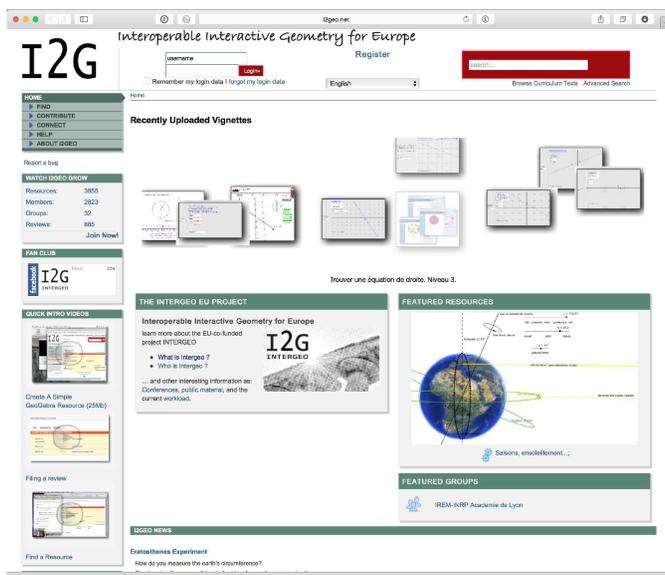
Figure 1: The angular bisector used repeatedly. On the right the point B has been moved once around point A, leading to another instance in the configuration space of the construction.

## 2.1 Experimental Mathematics

The first learning scenario described in [1] is doing Experimental Mathematics, where a DGS is used to explore not only one but many examples of a geometric construction. In particular, using a geometric locus it is possible to explore several configurations at the same time and to change the parameters for many of them simultaneously. This concept has been identified not only in Interactive Geometry but also as a general technique in Interactive Visualization by Victor [13]. However, as described in [3] it is necessary that the mathematical theory behind any system used to do mathematical experiments must be highly consistent and deliver correct answers even in unforeseen configurations. The approach of most geometry software is to start with the requirements for working on basic examples coming from secondary school classrooms. But as soon as circles are used in a construction the software has to handle both the problem of “vanishing” intersections and the problem of multiple intersections. Still, Cinderella is the only software that consistently and correctly handles such situations. It must be acknowledged that we still lack examples where this indeed harms the teaching with DGS, but this could be related to the fact that DGS are not used for free and creative experimentation that can lead to deep insights, but because they are used in teaching scenarios that are restricted to a very clear teaching goal. This asks for empirical studies on classroom practice with respect to DGS use and experimentation.

## 2.2 Mathematics on the Web

In 2003 we described how interactive Mathematics can be represented on the web. The last years have shown that this is a very important direction, in particular with the trend of Open Educational Resources (OER), that is, free to use, modify and redistribute material for teaching and learning. There are numerous websites that offer interactive content created with DGS. I would like to mention three of them.



☒ 2: The Intergeo portal ([i2geo.net](http://i2geo.net)) is specialized for content using interactive geometry software.

1. The Intergeo project [4, 6] (Fig. 2) created a web platform where interactive content can be collected and categorized according to topics, competences and educational levels.

Intergeo solved some problems that occur in the context of searching and finding resources, in particular with respect to multilinguality. All resources can be equipped with metadata that is language-independent and reflects topics and competences that are mentioned in curricula. Also, resources can be rated using a questionnaire that builds on didactical criteria for interactive media in teaching. Still, the platform has not been widely adopted, probably due to the additional overhead when adding material and suboptimal usability when searching for content.

2. These two issues are solved with GeoGebraTube (<http://www.geogebraTube.com>), which is tightly integrated into the GeoGebra Software. You can publish constructions from GeoGebra directly to GeoGebraTube.<sup>2</sup> On the platform they are available without prior installation of the software, not only for Java-enabled browsers but also for modern browsers using HTML5. In particular, these constructions

<sup>2</sup>Actually, this is now the only supported way to publish GeoGebra constructions on the web, as the original HTML export has been removed. This is a questionable move, as the terms of use and the privacy statement of GeoGebraTube have to be accepted if you want to publish your work there or embed content from there in your own pages? which is not compatible with the Open Educational Resource idea.

can be used on tablets as well (see also Sec. 2.6).

The approach behind GeoGebraTube is community-based: Anybody can publish his or her work, and everybody can comment on and rate resources. This allows for a huge amount of material (about 140 000 resources as of January 2015), but the collection lacks structure and coherence. This shows the strong connection to a platform like YouTube: You can find millions of videos there, but the only structure is given by *channels*.

3. Mathe Vital (Fig. 3) follows a different approach than GeoGebraTube. This collection has been created by a small team lead by Jürgen Richter-Gebert, and it is neither possible to add your own content nor to comment on resources. This restriction leads to the fact that the content is of very high quality and consistent, both in design and in mathematical terminology.

Mathe Vital and GeoGebraTube show two extremes on a scale: A large community can create thousands of resources, but the diversity of these resources makes it difficult for the user to use the content to learn without proper guidance. A small team can only create small collections, but these might be better suited for unguided learning. There is no reason to prefer one approach over the other – for both approaches there are use cases, and depending on the actual demands either approach can be better than the other. It is interesting to see that these two approaches seem to be a common theme in today's world: For all types of resources (news stories, videos, music, recipes, ...) there exist platforms and portals that either offer a huge amount of resources where you can select from using various criteria, or curated collections that consist of carefully pre-selected or manufactured resources.<sup>3</sup>

## 2.3 Interactive Exercises

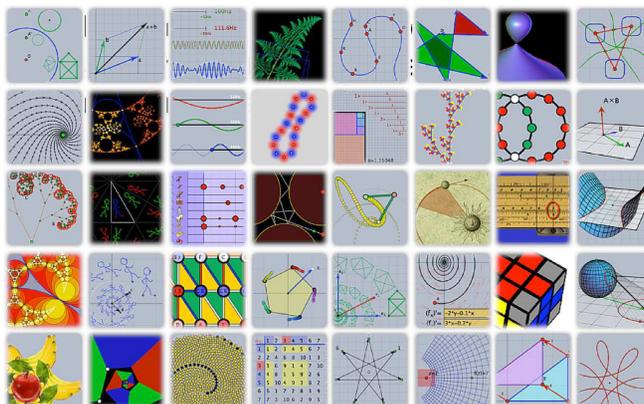
Interactive Exercises have been a major feature of Cinderella since 1997. However, these have hardly been used and the feature was more or less removed in the latest version. In our view, there are two reasons for the lack of adoption of this feature: (1) It was always difficult to use the built-in Exercise Editor for creating good exercises, and (2) in teaching it is difficult to find situations where it makes sense to use such self-checking exercises with automatic feedback.

During the last decade Learning Management Systems (LMS) became more and more popular. A good example is Moodle<sup>4</sup> that has been created in 1999-2002 and is now one of the most popular LMS worldwide. Such systems allow for the inclusion of interactive content, and we see this as a chance to revive the interactive exercises in Cinderella.

---

<sup>3</sup>You are invited to find out of what type the following portals are: Twitter, Google News, Spotify, TV stations. Find more examples!

<sup>4</sup><http://moodle.org>



☒ 3: The Mathe Vital portal ([www.mathe-vital.de](http://www.mathe-vital.de)) offers a huge collection of high-quality interactive visualizations. This is just an overview over some of them demonstrating the consistent design of the various visualizations.

Here, we have a system that is capable of recording student's scores, so it makes sense to include such exercises in courses. First approaches for this scoring are currently implemented in the context of the MC Squared project.<sup>5</sup>

It should be noted that recently (September 2014) a Moodle plugin has been published that allows for setting up questions that can be solved and checked automatically using GeoGebra ([https://moodle.org/plugins/view.php?plugin=qtype\\_GeoGebra](https://moodle.org/plugins/view.php?plugin=qtype_GeoGebra)). Unfortunately, there is no advanced support for checking; the correctness of a construction just depends on a single boolean value that must be created by the author of the GeoGebra construction.

## 2.4 Interactive Whiteboards

Interactive Whiteboards are widespread in schools in many countries. All manufacturers deliver software for writing on these boards and drivers to work with standard desktop software. There are even many applications that are specifically designed to support mathematics teaching. Our experience shows that this software is usually not up-to-par with standard DGS systems. So, most teachers who use mathematical software on whiteboards at use them with the software they know from the Desktop (usually GeoGebra). Advanced approaches like e-Chalk have almost vanished.

A very promising alternative software is (or was, as it appears to be discontinued) Open-Sankoré, formerly Uniboard.<sup>6</sup> This software is able to include *widgets* based on the

---

<sup>5</sup><http://mc2-project.eu>

<sup>6</sup><http://open-sankore.org/en>

W3C standard<sup>7</sup>, so it is possible to use HTML5-based geometry software, even with persistent data storage. Still, there are no capabilities for cross-widget communication as in the API specification for widgets in the MC Squared project. Based on the observations above we claim that it is necessary to develop a common standard for the inclusion of interactive content in LMS (coming from SCORM and LTI) and in whiteboard software (coming from the W3C standard).

## 2.5 Sketching

The most promising approach to Sketching so far is Sketchometry, a web-based geometry software that uses gesture recognition for input (<http://de.sketchometry.org>, Fig. 4). This is different from the approach that has been implemented in Cinderella and that was presented in 2003, where we tried to automatically generate constructions from a drawing. In that scenario, users should draw exactly like they draw with a pen. The necessary relations between objects in a drawing are subject to sophisticated guessing of the software: A line that is almost parallel to another line should become a *constructed parallel*, a line that is more or less round and goes through three points should become a circle defined by these three points, but if it goes through one point only and there is a point in the center of gravity of that line than it should be a circle defined by midpoint and a point on the circle, etc. Such guessing does not work in some situations, so we implemented additional hinting by pre-selection of elements. If you tap two points to select them and then draw a point more or less in the center of these two points, then the software will be more likely to guess that you mean the midpoint of the two pre-selected points. However, we are missing data that could show whether this input method is suitable for the use in the classroom, in particular the necessary devices were not widely available ten years ago.

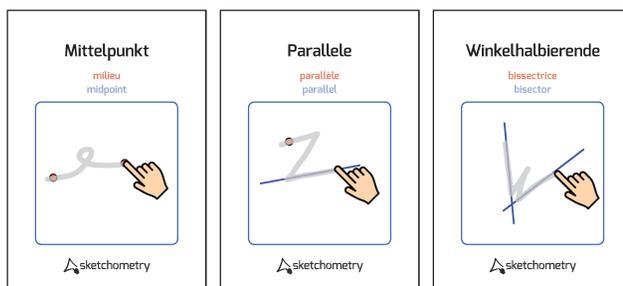
Compare this to Sketchometry: Here, a gesture for the midpoint is used, a line segment that contains a small loop to indicate that instead of the segment, the midpoint of that segment shall be constructed. So far it is unclear whether this approach is better or not to the interpretation of drawings with pre-selection, but the research team at the University of Bayreuth is currently investigating this in a school.

## 2.6 Mathematics on Mobile Devices

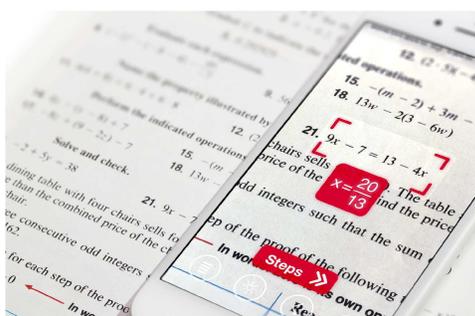
Mathematics on mobile devices is a rapidly growing area of interest. While there were first devices capable of high-resolution color graphics in the beginning of the century, we now have mobile phones and tablets that are superior to the laptop computers of that time. Actually, we can assume that within the next ten years every student in developed countries will have access to a mobile device that is powerful enough to do all

---

<sup>7</sup><http://www.w3.org/standards/techs/widgets>



☒ 4: Example gestures in Sketchometry. Source: Documentation of Sketchometry at <http://de.sketchometry.org/download/gesture-a4>



☒ 5: PhotoMath is an app that solves maths problems from books simply by taking a picture of them. It not only shows the solution, but it is also possible to get step-by-step explanations. Source: PhotoMath website <https://photomath.net/en/presskit>

school mathematics. A stunning example for what already is possible is the PhotoMath App (Fig. 5).

A technological drawback is that due to political reasons the universal Java platform was abandoned for mobile devices. Although Java was meant to be “write once — run anywhere”<sup>8</sup> and there were efficient Java Virtual Machines running on early devices (for example, on Windows CE or the Sharp Zaurus), this has not been pursued. Instead, Google, Apple and other large companies pushed JavaScript, which is available in Internet browsers both on Desktops and mobile devices. The concentration on this technology made it possible to have interactive content that is smoothly integrated in HTML pages and at the same time made JavaScript as fast as or even faster than Java.<sup>9</sup>

<sup>8</sup>See also [https://en.wikipedia.org/wiki/Write\\_once,\\_run\\_anywhere](https://en.wikipedia.org/wiki/Write_once,_run_anywhere)

<sup>9</sup>As a starting point I recommend reading <http://www.royvanrijn.com/blog/2012/07/java-speed-of-math/>, which points out that even in 2012 JavaScript could be faster than Java when used out-of-the-box, which is in line with our own experience.

However, most interactive academic educational software has been produced in Java so far. Currently, many software developers switch to JavaScript/HTML, either by starting from scratch or using compensating technology like the Google Web Toolkit GWT. For example, GeoGebra is using the same codebase for both its Java and JavaScript version, while JSXGraph has been written in JavaScript as a replacement for GEONExT. For mobile devices, the software that has been written natively in JavaScript is generally faster and better adapted to the device. The GeoGebra App, for example, is very slow for real-world examples, while JSXGraph performs smoothly. Some more thoughts with regard to mathematics education software can be found in [5], for example. There is one important lesson to learn: It is even more important to design a good user interface for mathematical software on mobile devices than it is for desktop applications — and it is not helpful at all to just copy the user interface from the desktop to the mobile device.

For Cinderella, we decided not to just translate the software from one programming language to the other, but to start from scratch in JavaScript and provide a software library named CindyJS that includes the major technology necessary for running Cinderella-based examples [14]. This includes the scripting language CindyScript, with a full compiler, the physics simulation framework, and the geometry kernel including complex tracing. All this is available as Open Source Software at <https://github.com/CindyJS>. Using these components, we can provide content that has been created in Cinderella via an export in interactive websites that work on any device, in interactive electronic books in Apple iBooks, as standalone apps for smartphones and tablets, or in creative books (c-books) as produced in the M C Squared project (<http://mc2-project.eu>).

Fascinating enough, it is possible to access modern hardware from CindyJS that has not been available to Cinderella before. In particular, it is possible to access the 3D hardware that is present in today's devices, allowing for extremely fast 2D and 3D rendering. Using the easy-to-learn CindyScript language it is possible to execute code in parallel on the graphics processing unit (GPU) [8].

## 2.7 Collaboration

As for collaborative work the disappointing message is that this aspect seems to be neglected currently. Although technology like multitouch interactive tables, fast wireless networking with Wi-Fi, Bluetooth, and NFC invite for cooperation, the classroom reality is that either students work alone with their devices, or the teacher is using a device to present. Although many people are using shared documents (through Google Docs, iCloud/iWork or similar infrastructure) and work with them at the same time, this trend has not been followed by DGS users. The sharing of constructions is usually asynchronous and not simultaneously.

It seems that teachers and students prefer to “own” a construction – and there are good reasons for that. *Experiencing* a construction is difficult to do together. If two

students are moving elements of the same construction, then neither of them can know which action triggered the reaction of the construction. This is different with real objects – if two students work with a physical model they can feel what they did and what was caused by the other student. So, any further development with respect to collaboration (or cooperation) is probably tied to future developments that connect the virtual environment to real objects.

## 2.8 The Future — Back in the Year 2003

In [1] we mentioned four directions of future research. Let us check what has been achieved!

### 2.8.1 Visualization of Algorithms

In 2003, we imagined a system based on Cinderella that is able to run and visualize algorithms. Meanwhile, the Visage system<sup>10</sup> has been presented. With Visage it is possible to visualize (not only) Graph Algorithms and manipulate them visually. In [2] we describe how this can be used in teaching.

### 2.8.2 A School PDA

As pointed out in Sec. 2.6, mobile devices have developed enormously. Moreover, many teachers and even students already own these capable devices, like iPads or Android tablets. Actually this has rendered the vision of a School PDA mostly unnecessary. There is only one reason left to have specialized devices only for mathematics, which is the use in examinations. A problem of tablets and mobile phones is that they are too powerful and also have a connection to the Internet, so — at least in Germany — their use in examinations is prohibited. The ban on such devices is a matter of debate for the next years: In school and particularly in mathematics, students should learn to use proper tools for solving their problems, and mobile computers and the Internet *are* proper tools that are also used for solving problems in university mathematics. Instead of fighting these devices they should be embraced and the curriculum should reflect their use, leading to even deeper insights in mathematics.

### 2.8.3 Natural language input

The input of construction via written language was another Cinderella research project in 2003. After the prototype implementations this has not been pursued any further, as it did not seem relevant for day-to-day teaching. With the availability of spoken

---

<sup>10</sup><http://cermat.org/visage/>

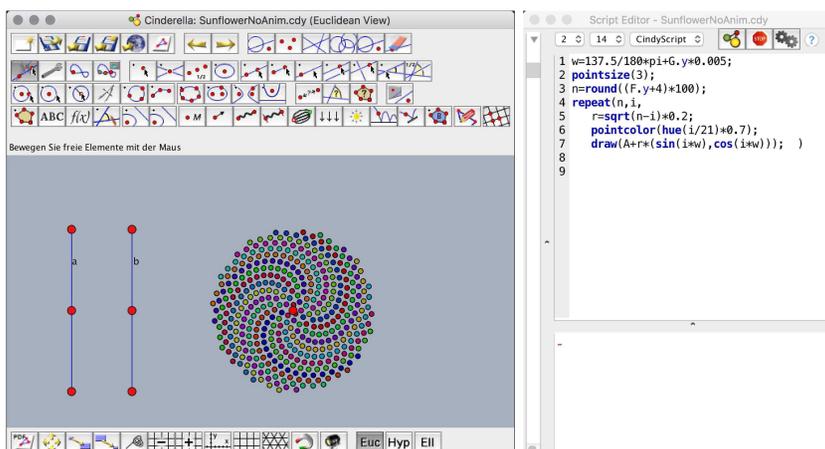


Fig 6: Some sample code in CindyScript creating a sunflower pattern

language input, for example through *Siri* on iOS (or the Speechkit library), it might become relevant again, in particular for handicapped users who cannot use a mouse or keyboard, but still want to work with Dynamic Geometry Systems. So, while there has been no progress in the last decade, we think that somebody should start working on such a research project again.

#### 2.8.4 Scripting and Macros

For exact formulations of construction sequences, for fine-grained control over the behavior of interactive figures, or for easier manipulation in animations, just to name a few examples, using a scripting language bears many advantages. Actually, what has been discussed as a future development in 2003 has been realized through the CindyScript language [9]. We refer to the full manual [12] here, which gives details to this language. Also, at <https://doc.cinderella.de> the full documentation is available online, in English as well as in Japanese, due to a translation provided by 入谷昭. One important project that has only become possible due to the available of the scripting language CindyScript KETCindy.<sup>11</sup>

As noted earlier, the CindyScript language can be used with the JavaScript implementation of Cinderella as well. It is even possible to write code in the CindyScript language directly embedded into HTML.

<sup>11</sup>see <https://sites.google.com/site/ketcindy/home>

### 3 Conclusion

In this article I gave a quick overview about the developments of Cinderella during the last decade, omitting many details and also many features that could be important (for example the interaction with the real world as described in [7]). Given all these exciting new possibilities that have been realized, are currently being realized, or can be imagined to be realized in the near future, we still have to raise the question what we are going to do with respect to mathematics education.

One important direction is to identify learning situations that might profit from these new possibilities. The process of curriculum development and the finding of new didactic ideas should not be ruled just by technical feasibility, but it cannot be detached from it. In order to know what and how to teach mathematics educators must know about the tools that are available. These can suggest new approaches and new topics. On the other hand, mathematics itself should stay independent of the tools that are available — the tools should follow the needs of teaching, not the teaching should follow the tools. So this has to be a dialogue between the tools and the teaching, to be held by maths education.

The other important direction that might have been neglected for a while is to work on the mathematical foundations of the software tools we are using. The (micro-)worlds created by software are something that is *experienced*, and this experience can only help learning mathematics if it matches the mathematical content, and the mathematical objects in these worlds should behave correctly.

### 参考文献

- [1] Enno Brehm and Ulrich Kortenkamp. Advanced Teaching of Geometry with Interactive Tools. In Ryosuke Nagaoka, Hideyuki Ishi, and Eckhard Hitzer, editors, *Proceedings of RIMS-Workshop ITMga 2003*, number 1378, Kyoto, 2004. RIMS.
- [2] Andreas Fest and Ulrich Kortenkamp. Teaching graph algorithms with visage. *Teaching Mathematics and Computer Science*, 7(1):35–50, 2009.
- [3] Ulrich Kortenkamp. Experimental mathematics and proofs – what is secure mathematical knowledge? *Zentralblatt für Didaktik der Mathematik*, 36(2):61–66, April 2004.
- [4] Ulrich Kortenkamp. Interoperable Interactive Geometry for Europe. *The Electronic Journal of Mathematics and Technology*, 5(1), 2011.
- [5] Ulrich Kortenkamp. Interaktives whiteboard, iPad & co. – das Klassenzimmer der Zukunft. Technical report, TU Dresden, 2012.

- [6] Ulrich Kortenkamp, Axel M. Blessing, Christian Dohrmann, Yves Kreis, Paul Libbrecht, and Christian Mercat. Interoperable interactive geometry for europe – first technological and educational results and future challenges of the intergeo project. In *Proceedings of the Sixth Congress of the European Society for Research in Mathematics Education. January 28th - February 1st 2009, Lyon (France)*, Lyon (France), January 2009. Proceedings of the Sixth Congress of the European Society for Research in Mathematics Education.
- [7] Ulrich Kortenkamp and Jürgen Richter-Gebert. Blended experimentation with dgs. In *Proceedings of CADGME 2009*, 2009.
- [8] Aaron Montag and Jürgen Richter-Gebert. Cindygl: Authoring gpu-based interactive mathematical content. In Gert-Martin Greuel, T. Koch, P. Paule, and A. Sommese, editors, *Mathematical Software – ICMS 2016*, volume 9725 of *Lecture Notes in Computer Science*, pages 319–326. Springer, 2016.
- [9] Jürgen Richter-Gebert and Ulrich Kortenkamp. The power of scripting: DGS meets programming. *Acta Didactica Napocensia*, 3(2):67–78, 2010.
- [10] Jürgen Richter-Gebert and Ulrich H. Kortenkamp. *The Interactive Geometry Software Cinderella*. Springer-Verlag, Heidelberg, 1999.
- [11] Jürgen Richter-Gebert and Ulrich H. Kortenkamp. シンデレラ. 幾何学のためのグラフィックス. Springer-Verlag, Tokyo, 2003.
- [12] Jürgen Richter-Gebert and Ulrich H. Kortenkamp. *The Cinderella.2 Manual – Working with the Interactive Geometry Software*. Springer-Verlag, 2012.
- [13] Bret Victor. Up and down the ladder of abstraction. a systemic approach to interactive visualization. <http://worrydream.com/LadderOfAbstraction/>, 2011.
- [14] Martin von Gagern, Ulrich Kortenkamp, Jürgen Richter-Gebert, and Michael Strobel. Cindyjs. mathematical visualization on modern devices. In Gert-Martin Greuel, T. Koch, P. Paule, and A. Sommese, editors, *Mathematical Software – ICMS 2016*, volume 9725 of *Lecture Notes in Computer Science*, pages 319–326. Springer, 2016.