



©1998 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the [IEEE](http://www.ieee.org).

[Return to Table of Contents](#)

---

## ***Educational Java Applets in Solid State Materials***

[C. R. Wie](#)

State University of New York at Buffalo  
Dept. of EE, 215C Bonner Hall, Buffalo, NY 14260

*[Invited paper, published in: IEEE Trans. on Educ. 41, No.4, p.354, Nov. 1998]*

---

*Abstract* - WWW-based coursewares will benefit from libraries of subject-specific software objects that are embeddable in html documents. Development of libraries of Java applet programs in various subject areas is timely and important. The developed applet library should enable educators to include these applet programs in their html-based multimedia courseware. This paper discusses the development and utilization of Java applet programs in the area of solid state materials and devices. The applet programs developed so far, with the purpose of providing a dynamic visual simulation of solid-state material concepts and device principles, are included in the courseware section.

---

<a href="#">Paper</a>	<a href="#">Courseware</a>
-----------------------	----------------------------

---

### **I. Introduction: Reuse of Educational Applets**

Reusable software objects, in the form of Java applet programs, provide an open, distributed and expandable platform for a courseware system. The applet programs, written in the object-oriented programming language Java which is developed and distributed by SUN Microsystems, inc., possess excellent characteristics for reusable educational software objects. The Java applets can be easily included in a html document along with other multimedia elements such as images, videos and audios, permitting an easy configuration of dynamic multimedia learning materials. This is due to the fact that the applet programs run embedded in a web page. Merits of Java as a programming language for developing courseware was discussed by Chirico, et. al. [1].

An important advantage of Java applets as educational software is that its reuse by educators is very easy. In order to produce a widespread impact, the ease with which to compose a courseware by educators and an easy use of the courseware by students is extremely important. The overhead involved in courseware, such as in learning how to include an applet in a courseware or in learning how to use the courseware itself, is virtually zero. Any educator with experience in composing a html page can easily

include an applet program into his or her html document. Learners simply need to visit the courseware page with a popular web browser in order to start learning using the applet programs. This is one of the greatest assets that Java applets provide for educational application. Other advantages include the platform-independence and the distributed nature of runtime environment. A html authoring tool could further facilitate educators and trainers to include the applet programs in their pages (slides, lecture notes, assignments, etc.). This authoring tool should be able to handle two possible sources of applet programs equally easily: applets from a local CD-ROM library and applets from a remote website.

For maximum use and productivity by educators, including an applet program in a html document should be as easy as including a clip art or an image file. A simple drag-and-drop operation could enable educators to put a high quality Java applet program in their lecture slides, tutorial notes, and homework assignment questions. This document may be served over the WWW from the educator's home page or from the course website. The main limitation at the present time is the lack of a sufficient number of subject-specific Java applet programs of a high educational quality and the lack of a html authoring tool with an easy-to-use author interface. It is important and timely to develop applet programs of a high educational quality on various subject topics of each specialty area. The benefit, namely, the learning effectiveness in learner's part and the productivity in educator's part can easily compensate the high costs involved in the applet development effort. An applet library in a given subject area, once developed in a substantial quantity and distributed, can form a base collection of that subject. This base collection can continue to grow via participation of both the academic and the developer communities.

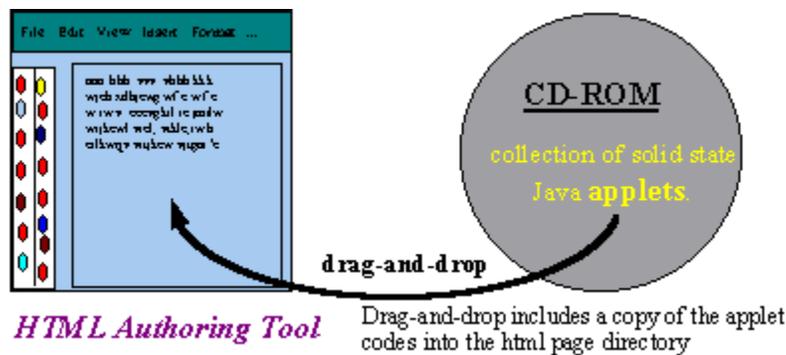


Fig.1 Drag-and-drop from CD-ROM into html page: physical copy of applet programs into new webpage.

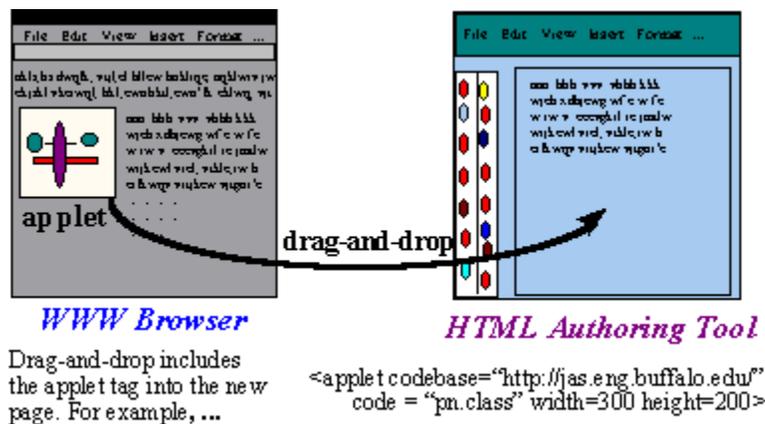


Fig.2 Drag-and-drop from a web site into local web page: just the applet link is inserted into new page.

Figures 1 and 2, shown above, are a model for a web authoring tool which is to enable educators to

employ Java applet programs with ease. Figure 1, using a CD-ROM library, is a form of physical dissemination of the applets. Figure 2 is a WWW dissemination of Java applets. For a WWW applet source, the html authoring tool should insert a remote link to the applet program in the html page by means of the *applet tag's codebase* parameter. This 'virtual' dissemination also allows the applet to appear as part of the educator's own lecture notes or lecture slides. The users (students) will see no difference in the educator's web page whether a remote link or a local link is used except perhaps for the internet downloading time of the applet codes. Subject-specific applet library and a drag-and-drop web authoring tool should benefit both educators and learners greatly.

## II. Subject-Specific Development Issues (Solid-State Materials)

This paper deals with development and utilization of Java applets in solid state material concepts, semiconductor device principles, and simple circuit principles such as single stage amplifiers and digital gates.

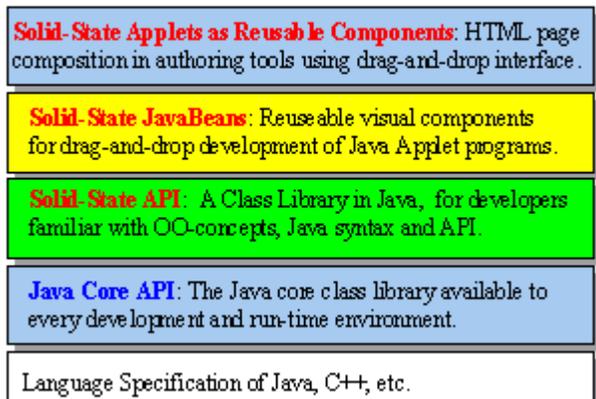
The purpose of our applets is to produce dynamic visualization of physical concepts for solid-state materials and to visually illustrate various device principles. Examples of currently available applet programs include *Fermi energy level concepts*, *energy band diagrams*, *current flow across a pn junction*, and *metal-oxide-semiconductor(MOS) field effect transistor (FET)*. Future development plan includes a MOS capacitor, bipolar junction transistor (BJT), single stage amplifier involving a BJT or a MOSFET. Emphasis is on an interactive visualization of important concepts that often escape student's attention and of those concepts that are difficult to develop a global picture based on the traditional learning materials only. A student in a physical electronics course calls it a "visual realization of theory".

Three layers of re-usable software components (or objects) could be developed. The three layers are (1) solid-state application programmer's interface (API) for Java programmers; (2) solid-state *JavaBeans*<sup>tm</sup> for application developers; and (3) solid-state applets for educators and trainers. In this project we are focusing on the solid-state API and the solid-state applets. Figure 3 shows, as top three layers, these reusable software components for the solid-state applet project, which in turn depend on the Java core class library (or Java API).

Fig.3 Various layers of the reusable software components which are the final products or an intermediate byproducts during the development process. In our project, the solid-state API and the solid-state applets are the two reusable components developed.

Java, as a programming language, is a pure object-oriented language. It is beneficial for the programming productivity that programmers follow an object-oriented design practice. By following an object-oriented design principle a class library in solid-state materials is being developed to assist the development process. The *solid-state class library* is developed in parallel with the development of solid-state applets and these library classes are improved along the way. Developed Java applet programs will eventually become part of a solid-state applet library which will be disseminated both in a CD-ROM and over the WWW. As was discussed earlier, this solid-state applet library will provide these visual simulation tools (i.e., the applet programs) as reusable software objects for use by educators and trainers (who are usually non-

### Layers of Reusable Software Components



programmers). Another important library of reusable software components is the solid-state JavaBeans (not part of our effort). The JavaBeans library may enable application developers to write applets based their own idea and imagination without spending so much effort as in writing applets based on the Java API alone.

---

### **III. Concerning the Course Use and Instructional Approach**

The html pages will usually be served from a course website, from an instructor's webpage, from a centralized website, or from a combination of the above. A CD-ROM applet library may be distributed with a textbook. As the student use of WWW can now be assumed to be a common practice, web-based instructional materials can be seamlessly integrated within a traditional course structure.

It can be seamlessly integrated with a traditional University course structure: lecture, recitation/lab, and homework assignment. Java applets, if used in a homework assignment or in a recitation session which is held in a computer lab, produce the best impact on the student learning as the student use of the applet programs is hands-on. For use in homework assignments, applets need be accompanied by an introductory text and an easy-to-follow worksheet which should be completed by students while working with the applet. For use in recitations, an instructor-led (or teaching assistant (TA)-led) walk-through of each applet program may be followed by student's hands-on simulation while completing the worksheet. Programs may be used in lecture presentations, but this produces only a limited impact on the student learning as the lecture presentation makes the students only passive observers. Lecture presentation may preview the applets before a homework assignment is given involving the applets. We have used the recitation approach which was composed of a TA-led walk-through of the applet programs, followed by student's hands-on simulation. The outcome assessment as measured from student feedback and test scores was very positive. This was reported in a separate paper [2].

---

### **IV. Concerning the Organization of Java Applets**

Simulation programs, interactive calculation tools, or graphing tools, etc. usually concern a set of concepts; therefore, an educational software component covers at most a few pages worth of materials of a textbook. Developing a collection of a few hundred Java applet programs focused on a coherent subject topic (solid state materials), equivalent to a whole book, is the aim of our project. A given applet program may be used in different applet books at different levels and in different disciplines. Once available in a substantial number of subject areas, these applet books may be organized in Dewey's decimal system in projects like Educational Object Economy of Apple Computer, inc. [3] or Gamelan's applet organization [4].

Within a book-equivalent sublibrary of applet programs, a well designed organization scheme will be important. Our main objective is to develop and disseminate a sufficient number of applet programs in the solid state materials area. The specific organization scheme needs yet to be decided. A systematic design approach such as the one used by Mengel et.al. [5] will likely produce more impact on both educators and learners.

---

### **V. General Discussion**

## 1. Java Applets as Educational Software

A developed Java applet program, which by nature depends on the graphical user interface for user input and on a visual presentation of the simulation results, is an excellent tool for teaching/learning abstract concepts and principles of science. Simulation processes and results may be visually presented to the learner in real time. A dynamically changing view of the visual simulation applets helps keep the learners attention at a high level. It is likely that a sufficient number of Java applet programs will evolve over time for each subject area as a library of reusable software objects. Copies of the library may be disseminated in a CD-ROM via publishers to educators. Eventually, the use of Java applet programs by educators should become as easy as the use of clip arts by ordinary users.

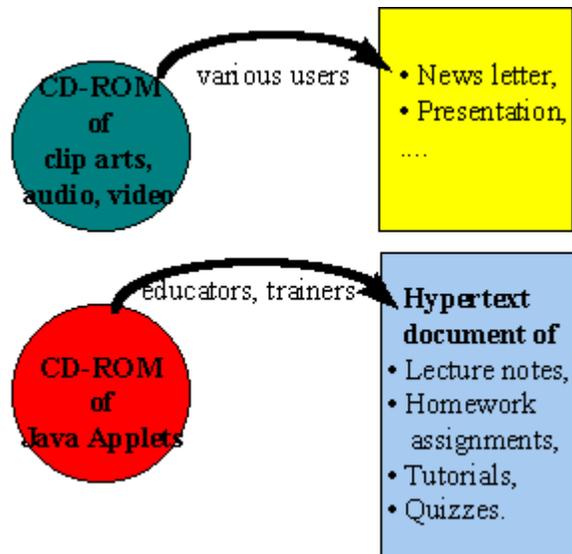


Fig.4 Including Java applets into html course documents should become as easy as including clip arts into ordinary documents.

An ideal development environment is an authoring tool with drag-and-drop architecture. If the originating source is a CD-ROM, then this operation copies the applet program into the same directory as the html document, and if the source is a remote website with the desired applet program, then this operation inserts an applet tag into the html page being edited. The difference between a WWW dissemination and a CD-ROM dissemination is that in the former, no Java codes are copied into the educator's local storage, where as in the latter the compiled Java codes are copied into educators local web sites.

## 2. Desirable Characteristics of Educational Software

Educational software systems may be categorized according to the mode of student access (*local* vs. *distributed*), its expandability by educator-users (*closed* vs. *open*), and the run time environment of the software objects (*stand-alone* vs. *embedded*). A courseware system should allow a distributed access and an easy expandability of the system by users (educators). This is readily accomplished if the software program runs embedded in a html document.

If we divide a courseware system between a *closed* system and an *open* system [6], then a prepackaged courseware system with platform-specific software programs written in a language like C is necessarily a closed system. Students may have access to the closed courseware either through a computer in the computer laboratory or via a CD-ROM purchased individually (e.g., with a textbook). This system is

rigid. An extension or reorganization by educators is not easy. A model for the closed courseware is shown in the right-hand side of Fig.5.

An *open* courseware system may be thought of as a collection of net-centric web-enabled learning materials or as an expandable courseware system which is compatible with an authoring tool such as SIMPLE [7]. Multimedia html courseware with graphical images, audio files, videos, and animations are an example of an open system. These multimedia elements are good expository tools, but have a danger of making learners mostly passive observers instead of active participants[8]. On the other hand, Java applet programs allow learners to interact more closely with the learning material, giving more control over the learning material to the learners. The simulation results are fed back in real time. In this sense the Java applets add a new dimension to the html-based multimedia courseware. They have a good property as expository and demonstrative tools [8]. An open system is open in the sense that a distributed access is possible via the WWW, new software objects may be easily added to the collection by the users (educators and trainers), reorganization and modification of the system are readily possible, and new learning materials may be easily composed by educator-users from a library of software objects. A model of an open courseware system is shown at the left-hand side of Fig.5.

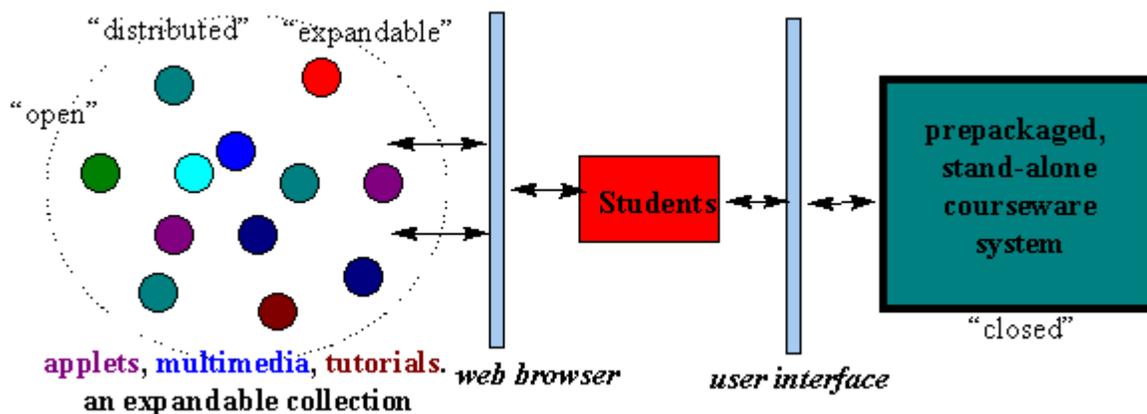


Fig.5 The 'open' versus 'closed' courseware system. Java applets possess all characteristics of an 'open' courseware system.

## VI. Future Direction

The applet codes and associated html materials in our solid state applet library need a more careful organization. We plan to use a formal design methodology, such as the method used by Mengel et. al. [5], in organizing our solid-state applet resource. Both a hypertext design methodology and an authoring tool such as SIMPLE [7] may be employed in the process.

## Acknowledgements

I gratefully acknowledge the support for this project by National Science Foundation under the grant no. DUE-9752316. Several students have helped develop some of the applet programs in the courseware section. Their names appear in each applet page.

## References

- [1]. M. Chirico, F. Giudici, A. Sappia and A. M. Scapolla, "The Real Experiment eXecution Approach to Networking Courseware", IEEE Trans. Ed. **40**, No.4, p.297, (1997)
- [2]. C.R.Wie, "Development of Java Applet Resources for Solid State Materials," J. Mater. Edu., **20**, No.1-2, pp.49-55 (1998)
- [3]. Educational Object Economy, <http://eoe.apple.com/>
- [4]. Gamelan, <http://www.developer.com/directories/pages/dir.java.html>
- [5]. S.A.Mengel and W.J.Adams, "The Need for a Hypertext Instructional Design Methodology", IEEE Trans. Ed., **39**, No.3, pp373-380, August 1996.
- [6]. A dichotomy between a *closed* system and an *open* system is purely arbitrary. A whole spectrum of coursewares should exist in between.
- [7]. W.M.Marcy and M.O.Hagler, "Implementation Issues in SIMPLE Learning Environments", IEEE Trans. Ed. **39**, No.3, pp.423-429, (1996).
- [8]. G.D.Bornida, D.Ponta, and G.Donzellini, "Methodologies and Tools for Learning Digital Electronics", IEEE Trans. Ed. **40**, No.4, p.294, Nov. 1997.
- 

## Author Contact Information

### C. R. Wie

SUNY-Buffalo, Dept. of Electrical Engineering, Bonner Hall, Buffalo, NY 14260

Phone: (716) 645-1023

Fax: (716) 645-3656

E-mail: [wie@buffalo.edu](mailto:wie@buffalo.edu)

---

## Author Biography

**Chu Ryang Wie** is Professor of Electrical Engineering at State University of New York at Buffalo. He received Ph.D. in Applied Physics from California Institute of Technology, Pasadena, CA in 1985; M.S. in Physics from Seoul National University, Seoul, Korea in 1982; and B.S. in Physics Education from Chonnam National University, Kuangju, Korea in 1980. He has done research in MeV ion implantation, radiation damage, deep levels, and interface properties in semiconductor materials and devices; resonant tunneling diodes; and high resolution x-ray diffraction. Past and current research work involves high resolution x-ray diffraction analysis of epitaxial semiconductors. Current interests include the development of WWW-based dynamic visualization tools for physics and engineering education; and the associated software engineering methodologies.