



The

# ROSE BYTER

Apple Blossom Computer Club  
A registered Apple/Macintosh User Group

Mar '09  
still only  
\$2.00

## Next Meeting

**Mar 19, 7 PM**  
**American Legion Hall**  
**406 SE Oak Ave**

## Agenda

1. Meeting starts at 7 P.M.
2. Intro's of members and guests
3. Old business
4. New biz
5. Program: (See number 6).
6. Questions & (maybe)Answers

## iChat-ery

Apple has provided a very nice program to allow people to communicate pretty directly with their computers. If you have an Internet connection, it's easy and most likely free. You can type back and forth one-to-one or as part of a group. Of course, Apple is in the business of not requiring much (other than the spending of your money) to enable you to do such things, so you don't have to type. You can talk back and forth or join the cacophony of a group speak. And why limit that to audio? Just add the right webcam and you can become your



very own talking head on peoples' monitors.

So, the question becomes, "Why aren't you doing it?" As this is being written, my "Buddy List" shows that not one iChat buddy is on-line. Of course, the weather today is very nice and conducive to not sitting in front of a computer. In fact, were it not **2 -->**

## Considering Your Opinions

by **Jim McClellan**  
<mcclellan@charter.net>

A while back we sent out a survey to our ABCC members to find out what you thought about several topics. One was for ideas for future meetings. I'd like to let you know how Walt and I dealt with this reply to this question.

The reply one member gave was "How service providers actually work?" Our discussion on-line in text form has been edited a bit because of space issues.

**Walt:** Perhaps we could get an actual service provider to come and tell us?

**Jim:** Yeah, I guess we might check **3 -->**

## Bat Sized Moths Unleashed In Eatery

At ABCC's latest "Bored Meat" most attendees were were astounded by the sudden fluttering from the vicinity of Pete Lipshutz's wallet, as he opened it and uttered that magic incantation, "I'll get it."

Now, I'd like to believe that, in this recession, the reason we didn't have a "Bored Meat" to decide what not to do at this month's regular meeting was that we all intended to stick Pete with that check and duck. Sadly, that's not the reason. In a nearly terminal case of, "Waddayawannado? Idunno, waddayawannado?" it was decided not to decide – by default – where and when to have a "Bored Meat," thus avoiding the whole question of what to do for the regular meeting.

So, we're looking to actually have a "Bored Meat" prior to April's meeting. The most likely date is Saturday, March 28, 8 a.m. for breakfast at The Mark V. If their advertised breakfast special is anything like their regular breakfasts, it should be tasty and a pretty good deal. Who knows, someone might actually pick up the check for the coffee.

The **Apple Blossom Computer Club** (ABCC) is an Apple Computer Inc.. registered Macintosh and Apple ][ family user group. The ABCC publishes *The RoseByter* newsletter monthly which is posted to each paid up member and reciprocating user groups. ABCC participates in user group newsletter content exchange. The ABCC also maintains a WWW site at:

<http://www.abccmug.org>

## Membership

Just \$20/year! Send with your name, snail- & e-mail address & phone to:  
ABCC  
13748 Lookingglass Rd.  
Winston, OR 97496

## Current ABCC Leadership

### Treasurer

Jim McClellan  
<mcclellan@charter.net>

### Apple Ambassador

Jim McClellan

### Web Master

Jim McClellan

### AppleScript Guru

Jack Webster <jackw@rio.com>

### Newsletter Editor

Walt Pawley <walt@wump.org>  
Send your stories and newsletter ideas to the Editor, Walt Pawley, at <walt@wump.org>. Plain text files are preferred, sent within the body of an email message or as an attachment. Mail physical media to:

**676 River Bend Road  
Roseburg, OR 97470**

Please understand that materials submitted may not be used and those that are will likely be edited.  
Copyright© 2009, All Rights Reserved

## ←-1 iChat-ery

necessary for me to fill this newsletter with what maanderings I can, I'd be outside mucking about myself. But today's iChat Buddy List looks a lot like it looks on other days – deader'n a door nail.

Now I don't happen to have a webcam, much less one that's compatible with iChat, so I can't do video myself. And, I suspect I need to do some mucking about with my router to make it work anyway. Fortunately, most people won't have that problem. It'll just work, like Mac stuff is supposed to. But, you have to actually use it.

To do that, you'll need to get a "handle" from a compatible chat supplier: either Apple's **.mac** or America OnLine's AIM. Last time I checked, either could be had free, though doing does require a bit of hunting for the links on the their respective web pages. At <https://new.aol.com/productsweb/Controller.jsp?promocode=825028&ncid=txtlnkuswebr00000002>, you should be able to sign up for a "Screen Name" which is what AOL calls a handle. Don't be surprised if that's changed by next week. Persevere. This, <https://secure.me.com/wo/WebObjects/Sign-up.woa/wa/trial?aff=consumer&cty=US&lang=en>, may work for getting a **.mac** handle. In either case, you don't need to spend money though they'll give you plenty of opportunity to do that,

So, what is it that these services provide? For the most part, they behave a little like a telephone switchboard. When you're running iChat (or any other compatible chat program like Adium, say), the program logs you in to the server as someone who can be reached. It's a bit like plugging in your phone. The phone company can sense that you have plugged in your phone because it can sense the circuitry in the phone. So, when someone else is logged in and has your handle in their

Buddy List, they'll get an indication that you are available.

Here's the thing, some of the people in my Buddy List who actually do use chatting once in a while don't leave iChat running even though they're by their computer. They only run iChat when they want to contact someone. This is like unplugging the phone from the wall except when you want to make a phone call. Perhaps they really "vant to be alone." I don't know. But I suspect they might think that because they are logged in, they must be actively chatting. Not so. It is *not* impolite to ignore the fact that someone has become available in your Buddy List.

Another advantage of doing some work with iChat is that you can send files directly from one person's computer to another. There are other ways this can be done but the iChat method has some advantages. Today, it's not uncommon for people to attempt to send a full length video in email. Silly, yes. Uncommon: unfortunately no. The attachment idea in email was never intended to handle gigabytes at a whack. But these days, a single icon is all the hapless user sees, and it doesn't say anything about how much data is associated with it. Most email boxes have limits on the amount of data they can hold and it's frequently only 10MB or so. With iChat, you can set up the transfer and let it run, straight from the sender's computer to the receiver's.

If you get a chat handle, you can put it into your "sig" on your email. That way, people you correspond with via email can also chat with you. Sometimes that resolves things much faster than a back and forth email exchange which might take days.

To get started and keep in touch with fellow ABCCers, you can start with **trbeditor** and **jimmcmayor** as AIM buddies. Hope to see you there!



## <— 1 Your Opinions

which provider serves the most ABCCers.

**Jim:** Here are the email addresses of our members. Does that help pick a provider to ask to be our program?

**Walt:** Hang on - I'll "compute" it.

(Provided a list of 24 two level domain names including **character.net** with 4, **rosenet.net** with 3, four with 2, and 16 with 1.)

**Walt:** Unfortunately, a lot of people are NOT using their host's name for their email.

**Walt:** There are some obvious differences - like the media used to do the connections - cable, radio, phone line, etc.

**Jim:** Yeah, but the question was: How service providers actually work? Do the differences affect how person uses their connection?

**Walt:** Some things do. For example: spam policies.

**Jim:** I don't think the average ABCC'er cares about the technical differences, but the end result, then if they are changing their provider it will really make a difference.

**Walt:** Clearly I'm the wrong person to be advising about such things.

**Jim:** I wonder if this discussion would get more interest in the TRB if we included part in the next issue?

—

At this point if you are interested in "How service providers actually work?" why don't you come to our March 19th Apple Blossom meeting and speak up? If you are the member who asked the original question and for some reason can't attend the meeting, why not write a response for our next TRB issue? Or, if you just want to respond in the TRB send it to [TRBEditor@AOL.com](mailto:TRBEditor@AOL.com).



# \$7.2B for broadband Internet

The economic stimulus package allots \$7.2 billion for expanding broadband connections to underserved areas. The Agriculture Department's Rural Utilities Service will dispense \$2.5 billion and \$4.7 billion will go to the Commerce Department for disbursement.

The assistant secretary of Commerce for Communications and Information and the Federal Communications Commission will establish the Broadband Technology Opportunities Program and ensure that it doesn't conflict with other federal broadband initiatives and that every state receives at least one grant.

ARRA also charges Commerce with providing support, training and equipment to schools, libraries, medical

and health care providers, colleges, and other community organizations. It stresses support for job-creating facilities and public safety agencies.

FCC's National Telecommunications and Information Administration will dole out \$84 million in grants by October, \$756 million in fiscal 2010, about \$2.5 billion in 2012 and 2013, ending up by fiscal 2015.

Of the USDA's \$2.5 billion for broadband spending, ARRA specifies that at least 75 percent of any area that receives stimulus grant money must lack sufficient broadband service "to facilitate rural economic development as determined by the secretary of Agriculture."

## Quantum Computer Part Demonstrated

Richard Feynman proposed the idea of quantum computing in 1982. Now, scientists showed they could use a single atom to control another atom. With this effect, they hope to create working logic devices, similar to transistors, which could be used in a quantum computer.

The Johannes Rydberg's blockade principle predicts the wavelengths of energy emitted when an atom changes energy levels. Niels Bohr based his

Nobel Prize-winning atomic model theory in part on Rydberg's formula. Quantum theory tells us that energy exists in discrete packets, each of which is called a "quantum" of energy. Quantum theory explains how the world works on the scale of atoms, electrons and other subatomic particles. Objects this small behave in ways that defy everyday intuition.

For example, two ping pong balls coming close together have no affect

on each other. But when two particles come close together they can become entangled, continuing to affect each other long after they are separated. Entangled particles often have linked and opposite qualities, such as "spin." Entanglement—and another concept called quantum superposition—are why quantum computers should be capable of processing more information than digital computers.

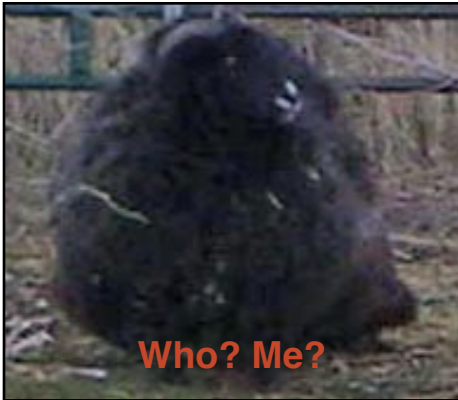


End-on view of high numerical aperture custom lens system used for trapping and imaging single atoms. (Image: Mark Saffman)

# Water Blogged Wump

Any trace of organization in these paragraphs is entirely coincidental

4



Who? Me?

.If you watch any commercial TV, you've probably seen the T-Mobile advertisement featuring a couple. The wife tells the husband that she's switched all their phones to T-Mobile. The husband is passively disgruntled about the lack of variety in his life. The wife retorts that they've been married 40 years and have had 11 bull dogs, all named Steve. I've probably had this played at me something like 100 times, most often with the sound muted so I don't have to listen to the sales hype. Today, I absently put the 40 and the 11 into my cranial computer and out popped the notion that **they must be fattening the bull dogs up to eat**. Or, at the very least, every three years, seven months and roughly 15 days, they trade their old bull dog in a new one. Or, perhaps my lack of experience living with bull dogs has kept the secret of their inordinately short lives from me. Of course, bull dogs are prized for the degree to which they're messed up compared to ordinary dogs ... if I've been informed more or less correctly. Maybe such short lives are "normal" in bull dogs. Somehow I rather doubt that.

I picked up the newspaper this evening and almost fell over from the impact of this part of the headline - **state managers take pay cut**. Wow! What a concept ... if you're approaching a state of brain deadness (you have read The Peter Principle, right?). In the article it says, "...more than 5,000 managers ... will save the state about \$6 million [over a] two year period..."

My initial reaction stemming from the lightning fast estimate of slightly over \$1,000 a year per manager (I glossed over the two year bit - which lowers the estimate to a whopping \$500 per year) that rattled into existence between my ears was, again, Wow! That was coupled with the more complex notion, "Doesn't anyone know how to do arithmetic anymore?" I mean, this is some real belt tightening on our "leaders'" part. For example, Kulongoski will experience a  $500/93,000 * 100$  % change; ie., 0.54 %. That has got to be tough to handle. Still, it's a bit better than most of those managers. They likely make somewhat less than \$93,000 per year. Some of them might experience cut backs as high as one whole percent. But this note is about arithmetic, not politics. According to another statement in the article, if the 27,000 unionized state employees would take the "same" reduction, the state would save \$120 million in the next two years. \$120 million divided by 27,000 is roughly \$4,444.44. I have to ask in what sense this is somehow the "same." Now I failed to do due diligence here. I didn't read the whole thing. I didn't research into whose numbers the reported figures happen to be. The article didn't contain a by-line. Political rhetoric is one thing. Arithmetic, is something else. Perhaps we should be doing a better job in our schools of training politicians and media people to do arithmetic.

I feel a bit like I'm crying, "Wolf!" I find the continual "upgrading" of software that goes on not merely annoying. I write about it routinely, but we apparently live in the dish detergent world in which it's more important to be "New and Improved" than it is to help you clean your dishes. Now I'm not really against upgrading software. In fact, it's an essential thing to do. What's wrong with it is how we go about it. It seems to be far more

important today to be always releasing some update than it is to actually fix real problems. Indeed, **it's quite common for upgrades to convert nicely operating computer systems, which do the bidding of their users just fine, into non-functional nightmares**. Considering the nature of software creation tools, this is hardly surprising. If anything, they change faster than the overlying programs people directly interface with. This would not be bad, if the changes were to fix bugs - some are, but most are adding "new features." And, indeed, every once in a while, the powers that be decide that it's a good thing to simply switch the underlying mechanisms from what they've been to something completely different ... "because." This necessitates their building a bunch of compatibility "scaffolding" software or simply telling their existing clients to suck eggs. On the Mac, there are multiple layers of scaffolding holding up massively complex networks of computer code - sometimes I think it may rival the complexity of the Federal Income Tax Code. Sadly, most of the people who write software seem to revel in wending their way through monumental complexity. I suppose they feel like this somehow sets them apart from other mere mortals. Perhaps there is some value to this situation but I'm skeptical. I much prefer using simple, consistent tools which I can use to accomplish what I desire. These days, very few such software, or hardware for that matter, environments exist. That is not a necessity. It's just the way it's gone and shows no sign of abandoning course any time real soon now.

Something else I write about routinely is light pollution. But it's a rare event - unique if my memory serves - when my wife suggests that I write something for this column. This she did this very evening. Earlier in the day, the March 2009 edition of National

## <--Water Blogged Wump

Geographic was delivered to our mail box. It's one of the few publications we treat ourselves to. In this issue, there are numerous reader letters about the November 2008 issue which contained an article about light pollution and its evils. I didn't write about it, either here or there – it's that “crying wolf” syndrome. One of the things that's certain about us as a people is that **our national attention span is fickle and most probably way shorter than it needs to be to accomplish any real improvements ...** even assuming we could come to something like agreement on what constitutes an improvement. It's pretty obvious that vested interests, at the very least, have convinced the collective us that lighting up the night is a really great thing. One of the nine letters waxed eloquently about how flying over the country at night with all the lights shinning brightly was an expression of just how far we'd advanced technologically – particularly about lighting up runways. If the poor schmuck understood anything about perception he'd probably be scared out of his wits by just how hard it is to locate the lights of an airport in the middle of a well lit city. While I'm quite certain power and fuel company executives would be very upset, we could readily “green” our environment while improving our economy by making some serious changes to throttle back light pollution. The very fact that we can see huge amounts of light from above cities is a dead giveaway that we're simply wasting energy. If that were all we were accomplishing, it would be reason enough to begin changing things. But we accomplish much more with our insistence on lighting up the night: not much of it which I think of as good. I'd like to be able to see stars at night. I'd prefer that my neighbors' automatic yard lights didn't make it easy for trespassers to wander around in our yard at night. I'd like to be able to drive down the road without seeing peoples' driveway lights looking

like cars ready to broach my path. I'll cease boring you with my list.

One is hard pressed to seriously claim that the computing business is not trying real hard to help bolster the economy. Just think back across the years: **when was there a time when the computer business wasn't all about “new and improved?”** Somewhen, about a decade or so ago, the overwhelming majority of our real computing needs were well satisfied with the technology of the day. Had nothing changed in the computing hardware of the day, we'd mostly just have extremely reliable means of doing our work well. Despite the fact that a really fast typist hasn't been able to outspeed a decent text entry program for decades, we keep getting faster and faster computers ... largely on the theory that faster computers mean faster production. For some work, this is true but not for what the majority of us do. These days, the main thing most of us deal with that requires speedy computers is the rendering of web pages that are designers' wet dreams. Some of us deal with masses of digital photographs and are upset at iPhoto's lagging performance when dealing with 2,000+ images. We might think we need a faster computer to deal with that. But the real problem isn't the speed of the computer: it's how we're using it. Staying with the iPhoto example, it's a program that has to “know” about all the photos in its province and keep track of what you're having it do to them. I'd posit that most often, if you're cropping/editing a set of a few hundred pictures you brought back from a trip. you don't need to have the other couple of thousand pictures clogging up the works while you're doing it. But putting your collection into that state with iPhoto is not a “natural” thing to do and requires more effort. AppleWorks is like that too. It's natural condition is to retain a list of everything you've ever AppleWork'd. So, as time goes by, its need to muck about in what becomes a giant list bogs it down. While not obvious to the

casual user, this is easily remedied - ie. requires more effort. **5** All that additional effort is not really hard; it's merely something not obvious to the typical user. So, the programs we adopt because they're readily available are also things that tend to assist our demand for new, fast hardware. Of course, not everything is simple. Take video creation and editing, for example. It's been around for quite sometime (I know a gentleman who still maintains a G3 PowerMac to run OS 9 so he can use the video production software that runs there and died when OS X came along). The trouble with video is that it requires a much larger data set for one document than most things. But, video remains something less than everyone's playground - most of us don't do it and don't need to - but that hasn't stopped computer companies from selling us on it. Lately, they've managed to get onto a huge new potential: virtualization. If you follow the techno-press these days, every aspect of computing is undergoing virtualization ... for some reason. Even virtualization is being virtualized. In fact, Apple has been doing that for years in the case of first emulating their Motorola 68K based processors on the IBM PowerPC and then running that under Classic on Mac OS X – you can even take it another level by putting SheepSaver on an Intel Mac. So, virtualization is not all bad, I guess. But these days, it's being carried to what seem to me to be insane extremes. Corporations are being sold on the notion that they don't need to know where their data really is – it's on virtual volumes accessed over networks: the real devices can be anywhere that the network can reach. Were that all, it'd be OK, I suppose. But as Michael Swain puts it, “The user of this computer doesn't have to be the user of this computer.”

This evening I heard an interesting statistic for the first time – AIG has 600 trillion dollars (**I did not make that up**) of what I'll paraphrase as **6 -->**

# http://www.abccmug.org's New Undergarments 6

While it's nearly certain that no one has noticed, ABCC's web site (<http://www.abccmug.org>) recently underwent a major restructuring. While the content as rendered in a web browser has not changed all that much, the coding of the web pages has changed in a fundamental way. There are a number of reasons for making this transformation now. But we'll get into that.

There are two fundamentally different approaches to making web pages. For historical reasons, we use one that involves the creator of the web

page having some idea of what the technology is about. The other method involves a more typical Mac-ish approach of using some program to do page layout on ones Mac and then transferring the result to the web site hosting service and hoping that it will work. This latter method typically doesn't require any understanding of the technology involved; rather, one just makes things look the way they want ... like they were creating an image to be printed on a magazine page.

Why is it done "the hard way?" Once

upon a time, no one around here had high speed connections. Even telephone modem connections were very slow (some of us still have a 300 baud modems ... not that they get much use these days). Bringing up a typical "outside world" web page was easy: a single click and go take out the trash, have a cup of copy, mow the lawn, check to see if the page was rendered yet or not, check the mail, read the paper and check to see if the page was rendered yet or not. Patience was a virtue if you needed to look at that page. More often than not, 7 -->

## <--Water Blogged Wump

BS "assets," which also turn out to be liabilities depending on which way the wind blows. Lately it's been blowing with hurricane strength in the liability direction. Here's the part that I don't understand. \$600,000,000,000,000.00 is \$100K for every man, woman and child in the whole world. Now I don't know about you, but were I involved in AIG's bookkeeping, I think I would have considered this an untenable situation; at least a situation bearing some careful scrutiny (actually, I'd have been screaming at anyone who'd listen that what AIG was doing was downright crooked). In the days when Lloyd's of London was formed, insurance was something that made sense. It's principle function was to spread risk. Today, that's no longer true. Today, the function of insurance is essentially to skim the cream off the activities of the world. It's become something that simply raises the cost of doing anything. If this sounds like sour grapes, just consider how insurance operates today. Essentially very rich people band together to take money from other people in exchange for a contract that is supposed to indemnify the buyer against something. The objective is not one of peers sharing risk. It's essentially the same thing as running a casino: the house has odds that guarantee it will

be profitable (which is what's wrong with the economy right now – greed and complacency made the house forget that they need to compute the odds), ie. they take more money in than they put out – they make a profit for the rich owners. Insurance adds cost to the activity being insured, pure and simple. Now the foregoing applies to regulated insurance activities. AIG has managed to create its 600 trillion dollar mess by side stepping the regulations so there were no checks or balances on it. Perhaps we're due for a Manhattan version of Judge Roy Bean. These financial geniuses who've gotten us into this mess are way bigger thieves than the horse thief or cattle rustler that swung in Roy's noose. He could build his gallows on the 60-th floor of executive buildings and really hang 'em high.

Out of character to the typically random musings I scratch out here, this paragraph is more about our insane economic situation. Early on, when I heard that the problem wasn't just in the U.S., but was global, I was thinking that the cause must be the sudden collapse in U.S. demand for foreign manufactured goods was putting a crimp in the style of countries like China. Imagine my chagrin when I find out that **I'm almost, but not quite, completely wrong**. It seems that the U.S. is still top dog in the

pecuniary emulation biz (look up the writings of Thorstein Veblen such as "The Theory of the Leisure Class," a 19-th century economist who was loath to let a syllable go unused – he had our modern problems nailed). It turns out that in their zeal to out cheat the big American cheaters, financial institutions in other countries have been busy creating an even larger fraction of bogus money than we managed. I suspect it was possible because we got away with it, so it must be OK, right? This brings up a horrifying thought: what if we decide we need to do a "Lend Lease" bailout to keep the European Union from collapsing? As the last paragraph pointed out, a trillion dollars is apparently now considered pocket change. Soon our politicians will bandy about quadrillion dollar amounts that need to be borrowed from the future. Thank goodness most of this money is just numbers in computers. If it were printed on paper as cash, I don't think we'd have any trees at all. One benefit of that scenario is that the total money supply would become a fixed quantity, once and for all putting an end the idiocy that economies must always grow. As long as we're stuck here on the planet, our domain is finite which means that the real economy is also finite, no matter what number of monetary units we assign to it's value.



command-period was used to stop the thing so you go back to using your computer. So, not only was patience a virtue, designing web pages that were useful without wasting bandwidth was also a virtue. Today these sorts of considerations are much less important ... for most of us, so if you're up for being ABCC's webmaster, don't be shy – you can do it your way.

ABCCmug.org was always written with HTML (HyperText Markup Language), which are just text files with some rules for telling the browser that's rendering the image of the page what to do with the data imbedded between the rules. These text files were hand crafted – or should I say hand-craped-ed. One of the big difficulties with doing such work by hand is that changes made in one place have a tendency to not to be reflected in other places. It takes a great deal of personal discipline to maintain a high degree of consistency ... and that's no guarantee.

There were some guidelines set out for the “look and feel” of our pages at the outset. This is where consistency is handy. Two main things were thought to be good idea at the time. First, a left “menu” area would exist for navigating the site. Being a simple site, there's little need for things like pop-up menus, etc. All we needed were simple HTML links to the pages involved. One further “rule” was that no page should link to itself through the menu. So, the menu entry for the page you're viewing should not be a link – just plain text. The other bit of consistency was to be at the bottom of the pages where we have links to user group friendly companies and the like. The consistency required could be done using Copy and Paste ... but this means that a human being must remember to do it. When we add a new “sponsor,” as we just did,

**7** it means going through all the pages and fixing them.

Before making the most recent change, an effort was made to deal with the issues of the last paragraph by using JavaScript. JavaScript is what's known as a client-side scripting technology. This means that the code for the JavaScript is put into the text of the web page and sent to the browser, which then executes the JavaScript code to generate whatever effect it's supposed to generate. Unfortunately, not all browsers do JavaScript the same way – you've heard of the “browser wars,” right? But there were two excuses for using JavaScript. First, we were already using some JavaScript for a few things (and still are) with no complaints, so it seemed that the sort of mainstream things intended would not be problematic. Second, the JavaScript code can be put in a single file that is referenced by the web pages. What this means is that changes that apply to all the pages covered by the JavaScript code only had to be edited into the JavaScript file itself, thus solving some consistency issues.

But before trying to figure out JavaScript code for doing the left side menu in just one file, some other issues arose that made JavaScript not quite so attractive. One of those issues is that it was desired to have a contact page with which a person could communicate with the group. Since this could involve lots of ridiculous input, it was decided not to send the messages directly to everyone in the group. Besides, that would have required maintaining a database of active users' email addresses on the web site – yet another source of potential inconsistency. But if you want to be an automatic recipient, just use the contact page to say so! Another issue was that it was of interest to get members' opinions about some things, so a web survey seemed a good approach (BTW: if you've not filled out your web survey, please do – your opinions matter and are taken seriously – if

you've lost the link to your survey page, use the contact page to say so). But this was a survey of members, not just anyone messing around on the web site. So, we didn't want to put any links to it. That's why you got email with a custom tailored URL to your survey web page.

In the effort to make these things possible, it was noticed that our web site hosting service provides PHP server-side scripting. PHP is a scripting language, originally targeted at supporting HTML but now used for other things as well, that runs on the computers containing the source text of ones web pages. When your web browser sends a request to get a page with a name that ends in **.php**, instead of that file's data being directly transferred to your computer, it invokes the PHP interpreter on the site which treats the file as a program written in PHP and the output of that program is what gets sent to your computer. Fortunately, most PHP code for simple stuff like ABCC's web site looks very much like the HTML code which it ends up outputting, making the whole process straight forward.

Like the JavaScript referred to above, the “boiler plate” PHP code can be maintained in a single file and simply included in the text of a page where it's needed. So, once again, we have the write-once, use-many-times capability that aids consistency. It turns out that PHP is a bit easier to read than JavaScript when doing something like the left side menu or rolling out of lot of mostly static page data. Perhaps most importantly, PHP does not depend on the browser being run by the user to do its work. Of course, one can still generate web page code that doesn't work “right” in some browsers. This is one of the main reasons we try to keep the pages simple, providing yet another sort of consistency – they almost always just work.



# Self Assembled: Ten Trillion Bits Per Square Inch



<http://newscenter.lbl.gov/feature-stories/2009/02/19/ten-trillion-bits/>

from Paul Preuss, (510) 486-6249, [paul\\_preuss@lbl.gov](mailto:paul_preuss@lbl.gov)

Self-assembling block copolymers could vastly improve the properties and manufacturing processes of nanostructured materials. Using crystals as a template, researchers have formed perfect arrays of nanoscopic block-copolymer structures extending over several square centimeters.

The magnetic domains that encode bits on today's computer hard drives a few dozen nanometers – billionths of a meter – across. Imagine an array of bits each just three nanometers across, ten trillion of them in a square inch.

Ting Xu of Lawrence Berkeley National Laboratory's Materials Sciences Division (MSD) and Tom Russell of the University of Massachusetts Amherst devised a method to make 10 trillion bits (10 terabits) – by assembling themselves. The researchers describe their work in the 20 February 2009 issue of the journal *Science*.

Xu pointed out that Tom Russell's laboratory has been working on polymer-based self-assembly for 20 years, concentrating for the last 10 years on one of the field's great challenges, long-range order. Long-range order means producing an array of nanoelements covering square centimeters or more, where each element can be addressed.

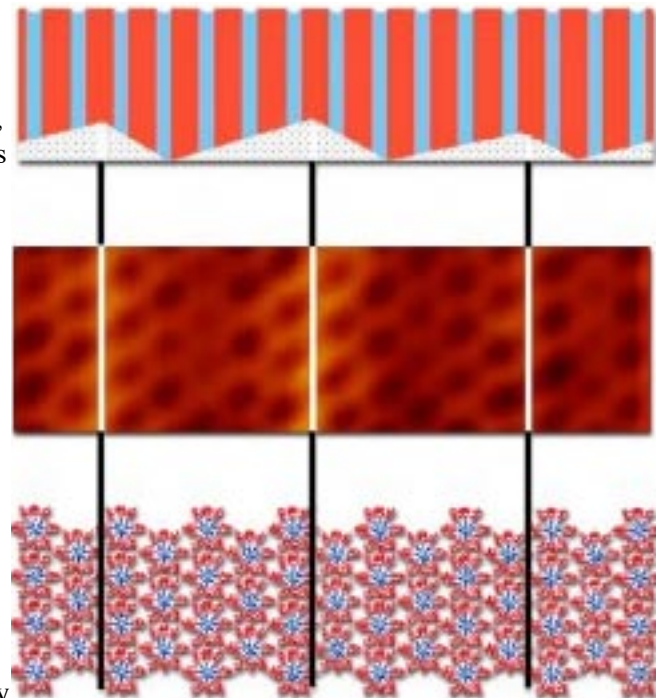
The initial steps were simple: large single crystals of sapphire were cut at specific crystallographic planes. The featureless cut crystal was then heated to very high temperatures, 1300° to 1500° Celsius (ca. 2400° to 2700° Fahrenheit) and annealed for 24 hours. During heating and annealing, atoms exposed between the edges of the cross-cut planes rearranged themselves in the lattice, with the result that the surface of the crystal reconstructed

itself as a series of parallel ridges, a sawtooth topography resembling, on the microscopic scale, that of the Western U.S.'s Basin and Range.

On this serrated surface, block copolymer thin films were allowed to self-assemble into nanoscopic cylinders standing upright from the surface of the sapphire. In microscope images the cylinders appear as hexagonally packed dots with arrangements readily directed by the parallel ridges of sapphire. Each cylinder in the array is a mere three nanometers in diameter, but the array extends over several square centimeters without a flaw, unguided by any pre-existing lithographic pattern.

"At first we were concerned that defects in the sapphire substrate could destroy the order of the array," Xu says. "What we found was just the opposite. Although there are indeed many dislocations in the surface of the annealed crystal, the self-assembling film of copolymers maintained its perfectly hexagonal organization right over them, covering an area of a few square centimeters."

"The challenge is to get materials to talk over multiple scales," says Xu. "An ideal scenario would be to throw a bunch of nanometer-scale Legos with all the properties you want into a box and shake it up. Let the molecular Legos assemble themselves into functional units without any complicated chemistry or processing."



An overhead view (bottom) shows cylindrical block-copolymer structures, consisting of a central polymer (blue) linked to a surrounding polymer (red). An atomic-force microscope image (center), shows the densely packed cylinders, dark in the center, with the varying height of the surface beneath them visible as alternating lighter and darker stripes. The side view diagram (top) shows how the cylinders arrange themselves along the ridges of the crystalline facets.

## unClassifieds

Need a manual?

Discount for ABCC members!



Apple Blossom  
Computer Club

<http://www.abccmug.org>

Give it a look.

Put in your own...

