Notes.net Iris Today

Face to face with Notes designer Bill Andreas

Interview by
Laura
Rutherford

Level: All
Works with: All
Updated: 07/02/2001

As the product designer for the Notes client and Domino Designer, Bill Andreas has witnessed first-hand the evolution of the Notes and Domino user interface. Here, he talks about how and why the UI has changed over the years—in particular, what factors influence the design and what role the Internet has played. Bill also shares UI plans for Rnext and discusses what factors may drive user interface design in the future.

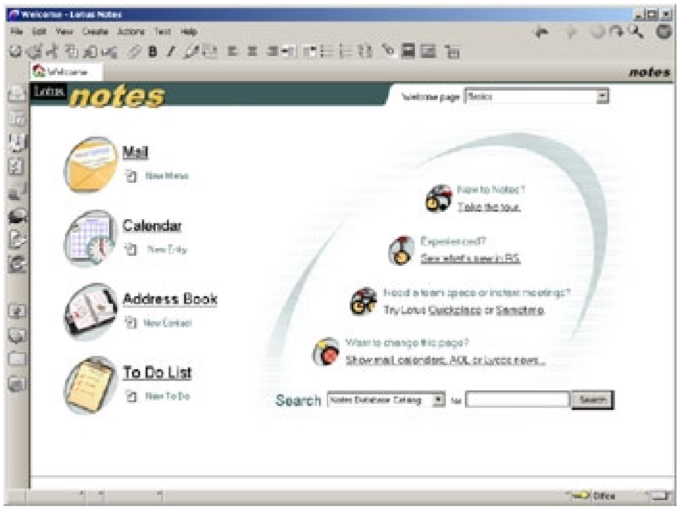
The Notes client and Domino Designer went through a complete redesign for R5. Can you summarize some of the major changes and reasons for those changes?
The most visible change on the R5 client was that you no longer saw the desktop when you started it up. Instead, you got a welcome page that was somewhat configurable—and we added bookmarks and tabs. There are lots more things that have changed, but for most users, the obvious thing is that when you start Notes, you don't see a bunch of chiclets. As a result, the basic paradigm for how you go about finding things and doing things in Notes changed.

Bill Andreas

There are a bunch of reasons why we made these particular changes. I think the biggest reason is that users, because of the Internet, no longer thought of things as databases. People don't think of things in terms of "Well, I have to go and use the SPR database." They're used to clicking on the URL to go somewhere. And clicking on a big thing that looks like a chiclet is very different in a user's mind than clicking on a URL.

Also, being able to bookmark anything, anywhere is what users of the Internet are used to, so this too drove some of the changes. Another thing that drove some of the changes is that we no longer had to support older user interface platforms such as OS2 and UNIX. OS2 in particular really limited what the UI in R4 could do because the Presentation Manager and OS2 are very primitive in terms of their user interface capabilities.

Recapping, what really drove the redesign was that users think about the world differently now; they don't think about chiclets. But what made it possible were changes in the underlying technologies—in other words, what platforms we did and did not have to support.



The R5 welcome page

How have customers responded to the R5 Notes client and Designer user interfaces?

Most of the customers responded well—probably 70 to 80 percent—which for a big change in the user interface, is a pretty good number. Places where customers did not respond well were where we underestimated training costs, particularly in some large organizations. We didn't have the time, when we planned R5, to understand how long it took some people to roll it out and how long we would have customers using R3, R4, and R5 simultaneously.

The places where we met resistance were mostly very large organizations that had perceived a high training cost and who needed us or wanted us to support a large spread of Notes revisions. Those are probably the main cases where we had some pushback.

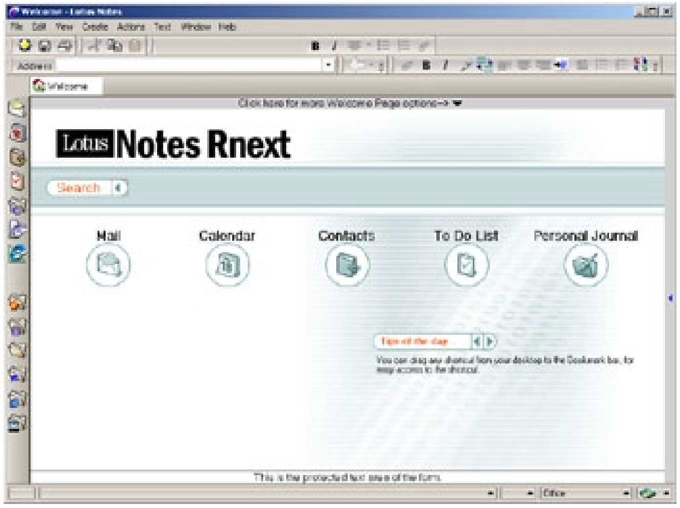
The other place where customers did not respond well (but on a lesser scale) was in response to some changes to keystrokes sequences. Power users were not used to the changes. We did try *not* to change a lot of those keystroke sequences; the ones we changed were really bugs.

Will the UI for Rnext have the same general look and feel as the UI for R5?

The look is bit different, but in terms of if a user wants to go do this or that—any one of a 150 or so common tasks—there is no change in where they click. There is no change in how they would go do something, other than that it might take fewer keystrokes.

When I say we've changed the look, I mean we've changed things like the icon for mail and the iconography for what the window tabs look like. While things may have a different look, they are in the same place. You still click in the same place and they still work the same way—although most of the things you are used to in the client have an awful lot more capabilities.

A good example of a UI change in Rnext is that we are replacing the old SmartIcons with modern style toolbars; so you can resize them, you can move them, you can put them here and there, you can have multiple lines of them. There's more flexibility in what you can have them do. You can have font pickers and font size pickers and a whole bunch of things. We have full-fledged, do-everything toolbars in Rnext.



You can drag any shortcut from your desktop to the Bookmark bar, for easy access to the shortcut.

This is the protected text area of the form.

The Rnext welcome page

Bookmarks have an enormous number of new capabilities for searching and sorting and filtering, and multiple ways of representing the contents of the bookmark. There is a way of eliminating bookmarks from the UI if you have a user who does not want to use them a lot or only wants a few simple ones that have big labels next to them.

We've also implemented a new display of bookmarks, which was intended for R5 but which we didn't get to because of time limitations. For any folder of bookmarks, you can now toggle back and forth between a "list-style" display (as in R5) or an "icon-style" display (which looks suspiciously like the old-style chiclets). It's been well known in UI studies for a long time that for some "explorer" tasks, lists work better while for other tasks, gridded icons work better. We intended to do both styles in R5, but as I mentioned, we didn't have the time. Both styles (lists and icons) have some new searching capabilities and better folder management/organization functions.

You mentioned toolbars and bookmarks as parts of the UI that offer some new functionality in Rnext. Anything else you want to mention?
Well, there is a lot. And we have tried to make a lot of it subtle. A simple example that an enormous number of users, especially power users, like is that a lot of our dialog boxes are now resizable. So if you do File - Database - Open and you need to make a dialog box bigger, you can. There are a lot of little things like that. I'm not even starting to touch on changes in the templates, such as Mail and Calendar. Those are areas where there are some significant new features that are pretty visible.

If we take a look just in the Notes framework in general, we've completely redone the Replication UI, and it's significantly simpler and easier to use. We've even redone things like the status bar. We are redoing user preferences so that you can get to all the user preferences from a single point. You can sort of look at it as: We've taken every element of the UI and looked at user feedback and we've put a lot of that in.

The templates are a little different. The changes that are being made in the templates vary from template to template. Mail has a lot of new features that have been added—not so much because customers have said, "We need these new features," but because if we watch usability videotapes, we see people trying to do certain things with mail. For example, a lot of people use their Inbox to manually manage the things they should be doing. If they get an e-mail that they know they have to respond to, they make sure that it stays unread so they can go back later and remember to respond to it. There are a whole bunch of new features going into the Mail template that revolve around people doing that sort of thing—how to make mail a way that you can actively track things you need to be doing, things that you need to respond to, and things you have responded to, and so on.

We are also redoing a lot of the Calendar UI. I know that some customers might not see it, but it is really one of the toughest user interfaces of any sort to do. Why? Because there are very few organizations that have the same requirements for what they need a user interface to do. So you have some organizations where no individual in the organization ever schedules a meeting; only the secretaries and administrative assistants schedule the meetings. You have some places where they don't have any concept of reserving a conference room. And you have some people who are continuously booking meetings across time zones. If we actually gave every feature that every organization really needed, the user interface would be enormous. But we have to support most of those features because there are large organizations that work this way or that way. We are coming up with what we hope is a rather innovative approach to some of these issues that lets every organization see just what they need to see.

How much UI training will R5 Notes users need when they upgrade to Rnext?
We don't know for sure yet. We always say none, but that is something we don't get a good handle on until we have most of the UI in place in final form. Our goal is that they don't need any, or if there is any training they do need, that the product is more than capable of giving it to them.

So what elements of the UI are you redesigning for Rnext? Why?
In the client frame, just about everything is getting new functionality to some extent. We have gone through just about every part of the UI. We have not necessarily changed the way it looks or the way works, but we may have put some more capabilities in there. For instance, take something as simple as type-ahead; it will now do a much more interesting "find." So we have a more sophisticated algorithm that is used if there is a long list of databases. Same thing is true in bookmarks. There is a much more sophisticated algorithm that tries to do that.

The same thing is true for all of the templates. We've gone through just about all of them. That doesn't mean that the templates will look much different, but they may have a lot more capability. Again, some things we are just trying to make intuitive. We hope a lot of users won't even notice the changes are there. For instance, with roaming user support, users might say "This didn't seem to do this before, now it does, isn't that great." I can't imagine that a user will need to be trained to understand that when he makes a bookmark on one machine, it shows up on all machines.

Where do you look for inspiration for the UI design—its colors, its graphical treatments, and its overall style?
Inspiration comes from different places. Part of it is driven by the simple fact that a lot of our users use Microsoft Office. We don't so much care about the look of Microsoft Office, but in terms of the feel, what you expect a toolbar to do, where you expect it to be—we do care about that. So we do look at Microsoft products and a few other major products we know a fair number of our customers use. And we look at the Web. So if we're going to redesign what our window tab looks like, we might make it look like Amazon.com because that's a metaphor that people know.

Styles are things that are amusing because they come and go. I mean, there really are fashions in UI design. It's a little more visible on the Web than it is in software. We always sort of think we know who our influential designers are. I think a lot of our ideas about simplicity of line—not so much color—are influenced by some of the major American architects like Frank Lloyd Wright. That's who we like to say is a major influence—and also some of the other designers from the 1920s and 30s.

In Rnext, we have added some colors that we would never have used in R5 simply because they are fashionable in user interfaces now. For example, orange is all the rage at the moment. But we don't go overboard on that; it's just that people who use the software everyday want it to look like, for example, Amazon.com.

We also use a lot of blues, in part, because I spend part of my summer working from a village in the Mediterranean where blue is such a dominant color—in the sky, the sea, and as the dominant color in the architecture.

What tools do you use to produce your designs?
Primarily two things: Photoshop and Notes. We have an advantage in that because we have the source code, we can make Notes do things that other people can't make Notes do. For certain kinds of things, such as animation, we do use some Macromedia products.

What factors influence the changes you make to the UI? For instance, current technology trends or needs, customer demand, and so on.
There are three things that drive it more than anything else. One thing is technology changes. Seven years ago, you could not make a graphic-intensive UI. The machine simply could not support it. Today, you can. Not only can you, but it carries virtually no performance impact at all. Yes, you could go overboard; but if you look at Notes, there is really no performance penalty for the graphics. So technology really does push it. We had to design for 16-color displays until a few years ago, and that certainly limits your user interface.

Technology also drives what people expect out of operating systems. People who are using Windows XP or who are using the latest Mac operating system have a different expectation about what the user interface looks like than people using Windows 95. Windows XP and the latest Mac operating system have much more subtle graphic effects throughout the user interface. So those users expect much more subtle effects, and that causes a problem because people who are still using Windows 95 and 98 are not used to those effects and wonder what all these weird things are in the user interface.

Our customers clearly drive the changes as well, and we are driven by where we think the product needs to go. Customers don't necessarily give us this information directly. They tend to say things like, "You need to do something," and we have to translate a vague request or an indirect request into what we really think is the needed feature. Some customers are very articulate and can specify exactly what they need, but in many cases, we have to listen to what a lot of different customers say and sift through it all to find the real, common requirement.

And there are some things we put in because we like to. And we think that is important. We are one of the only companies who put things in because we like to. Most other companies have a very rigid process for designing a user interface. We believe that any product used by as many people as Notes is, cannot have a single rigid process.

What type of research—formal and informal—do you conduct before designing or redesigning the UI?
We are all over the map on that. We have formal usability studies. We have formal usability teams that work on having people come in here and work on features—either on existing features or on new features under development.

We look at hours and hours of videotape.

We do on-site visits. We actually have people who go out and sit next to people using Notes in Anywhere, planet Earth; and they take notes and ask people what they think about things as they use them. We have in-house design efforts—right now we are looking at changes to the number and content of columns in the Mail Inbox view, for example. Somebody goes running around with little pieces of paper that show things. One of the nice things is that, in this building in Westford, we have people from all around the world, so when we use those examples, we know that we are internationalizing things pretty well.

There are very few things that would ever be put in the user interface that do not get at least some informal usability testing. Most of the major features and even the small ones are run through formal usability studies.

What role does usability testing play in influencing UI design?
We would love it if the role they played was to verify that we did absolutely everything right. That obviously does not happen. We go back and we redo the UI if it fails in usability. One thing that we can never be sure of is whether something passes usability here and then is going to fail in the real world. And that does happen. Sometimes we just happen to get a skewed sample of people or the way the lab works influences people.

We always keep a list of things after usability testing and ask “Do we have to do these things before this release or can it be redone in a point release?” Or in a few cases it may just be an unsolvable problem. No one has a user interface that deals with every issue well.

How do you design a UI for users who have different needs and different levels of expertise?
The goal is always to start by designing for the user who has the least expertise. Then what we try to do is think of the user interface like a screwdriver. Users who have limited knowledge of the product are going to look at the project to turn screws (that’s what a screwdriver does), but if you look at what tasks people use a screwdriver for, they use it for everything—opening paint cans, keeping a door open, piles of things. That is the way I try to go about making sure that experienced users or power users have a lot more capabilities. If the thing is supposed to be able to screw in a screw, good. A beginning user will be able to figure out how to do that. Power users can make that same tool do a lot of other things.

What are some of the visual techniques that help a user interface suit all users?
A good example is something like drag-and-drop. If you drag something from the obvious point A to point B, it copies it. But if you drag it somewhere else, it might do five additional things. The other thing is obviously to hide things in the user interface. Features are only available if somebody asks for them. The typical way to do that is to click a big button called, for example, More Features.

Most people over the years have learned to deal with making sure that beginning users or users who don’t care about anything else don’t have to use the features that power users or people who need more capabilities use. The power users can customize the user interface so that they can do more things.

In what ways does accessibility influence or affect your designs?
In R5, we made a conscious decision that we would support blind users and our support is pretty good—not perfect, but pretty good. We made another conscious decision to *not* do as much as we could have for supporting users with limited vision. There is a long history there. We basically are very unhappy with Microsoft’s support for users with low vision, and we did not feel

it solved a lot of people's problems. There are government regulations in place in this country and a lot of countries now. We are not happy with a lot of those regulations—not because we don't think we need to support those users, but because we think those regulations don't do a very good job of supporting those users—particularly the regulations that quote Microsoft. We think we could do a lot better. We have to meet those regulations whether we like it or not. And we will.

Supporting accessibility for handicapped users usually does not influence our design, mostly because we believe that all users need those features. It's not that we are adding features to handle accessibility. It's that all users need certain kinds of features. And in those regards, it affects the user interface enormously—but it is not for accessibility reasons. All users need the ability to control the color set used to draw the user interface because, for instance, the user does not like purple or pink, or the user is color-blind or has limited sight, or in the user's culture, green is a really bad color for a user interface.

We believe a good UI does not need an awful lot in it to specifically deal with accessibility. All good UIs handle a lot of those issues because they have to for all users. We try to use accessibility requirements to double-check that we are doing a good UI.

Can you give us your take on how the Notes UI has evolved over the history of the product?
When Notes first came out, they had to write the Windows code to give to Microsoft so you could even have a graphic. A lot of the UI has changed because users' expectations have changed. Users had trouble figuring out what a button was ten years ago; they had to learn you could click on it. These days, on a Web site, half the stuff does not even look like you could click on it, but people know to do that. Yes, that's not necessarily a good UI, but these days, users are far more knowledgeable about what computer user interfaces are. They are also more knowledgeable about what they don't like in user interfaces, which is good.

The scope of Notes has changed enormously. Notes does an enormous number of additional things than it did ten years ago. We try not to give a user interface to all that. We try to make a lot of that transparent. We support so many languages now—and with accessibility. I mean, if you look at the matrix, we have to support someone in Thailand who is blind.

The UI has also changed as the skill sets of people using Notes has broadened. It used to be that Notes was used by sophisticated users. Now Notes is used by people with different levels of skills.

More than anything, what has driven the evolution of Notes is users' expectations. I remember when we sat down in R4 and thought about "OK, what does a user think they need to click on." And back then, if it was not a big button labeled OK or Cancel, users probably would not think to click on it. User expectations now are that the software does not have boundaries.

Now that most Notes users are very Internet savvy, how has that vast Web use influenced the way you design the UI?
I think it has affected us in a great way. We are far less afraid of making mistakes. The idea that software is just this static box is not there anymore. I think that users expect that they can go anywhere, do anything, and user interfaces are far more graphical. Unfortunately, the people designing Web sites have absolutely no user interface experience. And those Web sites typically need to be seasonal—so if it is National Arbor Day, for example, they may have to have trees all over their site. You can't test your site continuously when you have that kind of stuff, so users have become more tolerant of that kind of user interface design. A lot of users will go through an incredible amount of pain to get something done. On the one hand, users expect more out of the user interface, but they also expect the user interface

to be bad.

Interestingly, what we have seen is that twenty-year-olds don't seem to have different expectations than thirty-year-olds. And that is very counter to everything you read everywhere. I think a lot of kids who spend a lot of time on the Web—I don't think that translates so much into a work atmosphere. The Web has affected older workers. It has made them more computer savvy.

How are the user interfaces for Notes and iNotes Web Access related? How are they different?

The goal over time is to make them as close to the same as we can. The big difference is that iNotes is running inside somebody else's client framework, mainly Internet Explorer. So they have to live within that and there are some things they can't do. Over time, we hope that the principal difference is within the frame—you get a lot more capabilities running the Notes mail inside the Notes client frame than you do with it running inside Internet Explorer. There are still difficulties. Browsers still have performance problems dealing with something as complex at the Notes UI.

What are the ultimate goals for a user interface? How do you work toward those goals?

The ultimate goal for a user interface is that it is not there. The user does not notice it. Will we ever get there? Who knows. I don't worry about that very much because when you look at those long-term goals of what should a user interface look like in the year-whatever, we are so far away from that state.

One of the things that does bother me—but we can't address it too much because the research is not there yet—is that users are now dealing with enormous amounts of information. People might think that is obvious, but that has changed from large amounts to enormous amounts. It has happened really within the last 18 months. People used to say we should have a paperless office, and e-mail really put more paper into the office. We actually are seeing companies now—without planning—who are truly becoming more of a paperless office. And that means that the amount of information coming in is enormous—not big, but enormous. That's an area for which we have to start finding answers. We are thinking about that a lot.

About Bill Andreas

Bill's been with Iris for about five years now, originally concentrating on the design of Domino Designer and the more geeky parts of the Notes client. Before Iris, he was cofounder of HyperDesk Corporation and the Object Management Group. When not working on the Notes UI, he's a dirt archaeologist working in the Mediterranean.

About Laura Rutherford

Laura worked as a user assistance writer for Lotus until she had her daughter, Kate, in January, 1999. Since then, she had another daughter (!) Maggie, born in September, 2000. Now Laura spends most of her time taking care of her two daughters, two dogs, and one husband (basically in that order). In her free time, she loves to read, run, and, believe it or not, write articles for *Iris Today*.