Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Contents

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

IE5 Cross-Platform Strategy

What is this strategy... ... and what isn't it?

For some time we have searched for increased cohesion and unity of purpose in delivering Internet Explorer across multiple non-win32 platforms. Not only is it time that we commit to a cross-platform story, but the reorg is affording us an opportunity to align our resources even more tightly with the plan. This document gives:

- clear product/business objectives for cross-platform 5.0 browsers
- the feature overview necessary to deliver consistent cross-platform browsers
- schedule for deliverables
- twilight plan for IE Mac

Cross-Platform Product Objectives

Background

Microsoft has now put out several versions of Internet Explorer on several platforms. While the win32 version of IE has continued to make serious strides in terms of functionality, and major inroads in terms of market share, the cross-platform versions have not made the same market share gains. While the lack of cross-platform market share is troubling, the negative impact on win32 IE market share is unacceptable.

While the various cross-platform development teams continue to put out quality products, win reviews, and add win32 compatible functionality, there is still a perception in the press and among key corporate customers and potential customers that Microsoft does not deliver a credible cross-platform web browser (note: this is different than delivering a credible Macintosh or UNIX web browser). As we talk to more and more customers, it is becoming increasingly apparent that the cross-platform browsers directly affect overall IE market share exponentially. The question becomes - what do we need to do to combat the perception (and in some cases, the reality) that we don't have a consistent cross-platform story?

The key tenets of achieving wide market share for all platforms for Internet Explorer 5:

1. Stability, Footprint, and Performance – This is not just an issue for the cross-platform browsers. In a market climate where customers have asked for fewer features that work better and faster, stability, footprint, and performance should be major priorities for all the browsers. The real challenge is not just improving this story for each browser, but being able to quantify these goals, and then communicate how we meet them to the outside world. Getting perceived customer value and impact from delivering on these asks will be a major challenge for our marketing team.

2. Sim-ship - Part of Netscape's success comes from the fact that they sim-ship their products (although key features are sometimes left out without any notice from press or customers). Each product benefits from being part of the overall launch effort. We could benefit as well from a combined (Win32, Win16, Mac, UNIX) launch, however marketing must be signed up for not only delivering marketing dollars and resources, but for helping us target the specialized customers on each of the platforms.

3. Focused, optimized messaging - Along with creating cross-platform browsers with a balance of core features versus platform specific features, we must have a marketing strategy and messaging that reflects that balance. As mentioned briefly above, only part of competing with Netscape means meeting them on their own ground (sim-shipping a product, evangelizing a core set of functionality). The other half means exploiting the weakness in their one browser/UI for every platform. This means that marketing must be highly educated about our different focuses beyond the core on each individual platform. This means our marcom must be sensitive to the different markets in which it is being presented. This is a huge challenge.

This means that while win32 will be the most feature-rich, if we focus on win32 to the exclusion of other platforms/markets, then we have negated the potential gains we get from sim-shipping that core set of functionality. Sometimes we have a difficult time balancing any message with our win32 story. We need to also remember that driving people to adopting IE as their web browser will eventually drive win32 adoption.

4. Attach, attach, attach

Each cross-platform browser needs to be in the position to attach to as many delivery vehicles on their platform as possible. On MacOS we are finally in the enviable position of being the default browser on the OS. This is a huge advantage especially in the case of the large chunk of customers who use whatever browser comes with/is integrated with their operating system. There are other opportunities as well.

Two of the main pillars of the win32 IE5 team's efforts are around adding value to Office9 and NT5 as integrated parts of their products. On the Mac for example there are similar opportunities with future releases of Office for Macintosh and MacOS. There are also many opportunities for integrating with other 3rd party products on each platform. Many 3rd party software developers aren't even aware of the flexibility the IEAK offers them in terms of deploying a customer version of IE with their software. This is a huge opportunity especially if we focus on the consistent deployment story for ISPs, corps and education. Much of the functionality bleeds over into being useful for ISVs.

5. Painless deployment - In the enterprise (both business and academic) the cross-platform versions of IE have the potential to impact market share exponentially. Making deployment as consistent and interoperable as possible across platforms is of paramount importance. It's difficult enough for IS managers to maintain alternate platforms (Macs for example) in an overwhelmingly windows-based environment. We don't want them to have to make any extra effort in deploying IE on those "minority" platforms.

6. Core DHTML is the web app/content platform - While still in its early stages, DHTML certainly has the potential to mature into a robust, high performance, stable platform for deploying web applications and independent web content. For developers of LOB applications as well as content providers, a cross-platform story is a necessity. Some level of XML support needs to be given serious consideration as part of (not as an alternative to) our core cross-platform web runtime as well.

Open Issues and Concerns

There are a large number of challenges that this strategy presents as well as many many details yet to be worked out. Here are some challenges we face:

- As mentioned above, all the benefits we get from sim-shipping a core subset of functionality are negated if marketing isn't prepared to have a very flexible set of messaging, strategy, and partnerships. This means a coordinated effort preaching about IE5 to different markets with different emphases. In a traditionally win32 focused marketing organization this will be an enormous challenge, especially as it pertains to cross-platform content development (as we've seen with channels in the IE 4 browsers.)
- The reality is, the cross-platform teams have a fraction of the development, marketing, and program management resources of the win32 team. Given any re-architecture work that needs to be done for each browser, there will be very little time for new feature work. In the IE5 time frame, we need to leave room for making interface and other visible feature enhancements (things that will be most visible to the end user), versus focusing solely on content rendering.
- While sim-shipping the browsers is going to be difficult, I suspect that sim-shipping public betas will be near impossible. (On the Mac, for example, we made the decision not to ship Preview 2 so we could reach our 3-month delta date.) Will the lack of coordination of betas detract from the benefits we are trying to get from sim-shipping with a consistent core story, or will it enable us to synch up on the final versions and thus have more impact?
- If we choose IE4 DHTML as the core cross-platform runtime story, not only will we have a difficult time evangelizing it versus the new IE5 win32 DHTML support, but we will probably have some

compatibility issues as well. If we follow this plan, the trident team will need to have some plan for backward compatibility.

- There are some separate deliverables for some of the platform teams that may affect overall ship dates. (For example, the Mac team needs to sync up with Apple's OS release schedule; we've also made commitments to Apple regarding features such as ColorSync support, etc. Right now, the next OS release appears to jibe with a potential sim-ship; however, if the schedule changes, we will have to reevaluate our situation.)
- We need to rationalize our cross-platform IE strategy with our OE strategy. I'm not familiar enough with the depth of issues presented by OE, but we will definitely need some clear coordination.

Feature Overview

Below are two potential feature lists. The first represents the bare minimum that must be implemented for the cross-platform IE5 clients. The second list contains a feature wish list that can be considered if the first list can be met on schedule. Note: some features have already been implemented on certain platforms and are listed as part of the cross-platform deliverables. While some features may have a particular platform listed as they are only applicable for that instance of the browser. (Note: this draft of the plan has yet to be reviewed by the Win16 and UNIX teams)

Minimum Bar Feature list

- **Runtime**
  - *Client Capabilities* – Although it is an IE5 DOM feature, the Client Capabilities OM is key to support in the cross-platform browsers as it enables content authors to get fine grain detail on what exactly the browser can and can't do.
  - *CSS 1* – consists mainly of cleanup work and testing against common test suites
  - *CSS Positioning* – consists mainly of cleanup work and testing against common test suites
  - *StyleSheet Object Model* – consists mainly of cleanup work and testing against common test suites
  - *HTML 4.0 (minus accessibility)* – accessibility is not a native Mac function. – consists mainly of cleanup work and testing against common test suites
  - *Scriptlets* – consists mainly of cleanup work and testing against common test suites
  - *Element Collection* – consists mainly of cleanup work and testing against common test suites
  - *Event Model* – consists mainly of cleanup work and testing against common test suites
  - *HTML Dialogs* – consists mainly of cleanup work and testing against common test suites
  - *Inner/Outer HTML* – consists mainly of cleanup work and testing against common test suites
  - *CDF* – consists mainly of cleanup work and testing against common test suites
  - *Tabular Data Control (TDC)* – This is a subset of the ADO functionality and should be supported and tested against a common test suite.
- **UI**
  - *Channel Bar* – this will be merged into the favorites bar. As no active desktop exists on the cross-platform browser, no cleanup is necessary.
  - *Search Bar* – As there will definitely be further evolution and enhancements to the Search Bar the cross-platform teams need to engage with the win32 Search team to ensure that these enhancements can migrate to the non-win32 browsers
  - *Offline cleanup* – This involves some refinements of the offline feedback in IE developed by the win32 team.
  - *Save As...* - the ability to save standalone documents of web pages or web sites (Mac browser uses WAF format, MHTML support listed in "would be nice" feature list)
  - *ICW (mac only)* – Apple has requested a major overhaul of our ICW UI to look more like their OS. This is key in directing Mac customers to our referral server
  - 3 cool new end-user things (coming in separate document)
- **TCO**
  - *Auto-proxy* – the ability to set the proxy via JavaScript on a hard-coded start page

  - *Static PID* – MTS needs to have a PID to monitor and control the tech support in the product
- **Services**
- **Architecture**

"Would be nice" Feature list

- **Runtime**
  - *VB Script (Mac only)* – the UNIX and Win16 browsers already support VBS. Putting VBS in the Mac browser consists of two major chunks of work, including porting the VBS engine, as well as making the browser ready for the nuances of VBS (case insensitivity, variable storage differences, etc.)
  - *CSS Transitions and Filters support or transparency* – the work on the actual filters is already done on the Mac, but they need to be hooked into the product. For each platform, filters need to degrade gracefully.
  - *DirectX Multimedia Controls (Mac only)* – a porting team has been working for several months porting DX3 as well as the multimedia controls to the Macintosh. This definitely is not part of a core cross-platform runtime, but a significant investment has already been made on the Mac.
  - *W3C DOM* – the Trident team still does not have a clear idea of exactly where MS stands in regards to the W3C DOM to come. While it is likely that the cross-platform teams will not have time to work on any W3C DOM compatibility, we should know where we stand in regards to this.
  - *XML/XSL, XML OM* – The XML team is working on a win32 XML/XSL engine. They have also made it a top priority to engineer their engine with sensitivity to cross-platform issues. However, it is unclear that there are more clients than CDF and OSD for these services in the short term. The long-term goal is definitely for third parties to depend on these services in the browsers. However this may be an initiative to pursue in the IE6 timeframe. In the meantime, each of the cross-platform browsers already has basic support CDF (although each implementation needs to be tested against a common test suite).
  - *Scriptoids* – The win32 team has come up with a new set of components called Scriptoids. They are a way to expose a script as a COM object.
- **UI**
  - *History by site (Mac only)* – this is a key end user feature that the Mac doesn't currently support
  - *Security Dialogs cleanup* – the win32 team has expressed interest in driving this. However it is unclear how much traction they will get against this priority. If they do succeed, the cross-platform teams should copy their efforts.
  - *Kiosk Mode* – the key to this feature is that it is configurable through the IEAK
- **TCO**
  - *Drag Install (Mac only)* – MacOffice has its own version of Darwin. This is key to support, but only if we can get in the Office folder slipstreamed onto their CD
  - *Dynamic PID* – MTS would really like the PID to be dynamic and different for each customer
  - *New Referral server* – the ICW needs to be integrated with the new referral server being worked on
- **Services**
  - *Native StuffIt Decompression (Mac only)* – incorporate licensed StuffIt decompression engine into IE
  - *MHTML generation and reading* – in order to make the "Save As..." feature work, consistently cross-platform, MHTML generation is needed.
  - *Personal Certificates (Mac only)* - do win16 and UNIX have this?
  - *S/MIME (Mac only)* – do win16 and UNIX have this?
- **Architecture**
  - *Factor rendering Engine (Mac only)* – the HTML rendering engine is used by both IE and OE on the Macintosh. It is currently not factored so that IE and OE can share it as a single library. Both for footprint considerations and to provide hooks for any future rendering engine to be built into Mac IE, this is key work that needs to be done in the IE5 time-frame.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Delivery Schedule

It is unclear as to whether a synchronized beta schedule is necessary in order to take full advantage of the simship message. More likely, Win32 IE's ties to the Office9 and NT5 betas make it extremely unlikely. Also, the diminished feature set in the cross-platform browsers may make 2 public betas unnecessary.

M0 – 3/1/98 – specs done, test teams coordinated
Beta – 6/1/98 – feature complete
Final – 8/15/98 – stability, performance tuning

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Twilight plan for IE Mac

Why Twilight IE Mac?

On the Mac specifically, Microsoft has put out several critically acclaimed and award winning releases. Yet our market share without AOL hovers around the 10% mark. Some key facts to consider:

- Mac market share continues to decrease making IE mac's market share even less relevant in the overall IE market share percentage
- Netscape appears to be decreasing their investment in the Mac
- Once we've delivered a core cross-platform consistent downlevel runtime, it is not clear what benefit we gain by bringing win32 technologies to the Mac
- IE 4.01 will be the default browser in the next release of MacOS which will probably not rev until mid-1999, at which point we may have gotten all the gains we're going to get

Continued long-term investment in development resources for Mac IE does not make sense. If we deem it valuable we may decide to invest marketing resources possibly with a minimal maintenance team to get the most from our investment. Variables that might dictate increased investment include:

- How important the Office team deems IE as part of their Office9 message
- Apple's reaction

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Releases

IE 4.0

There was a problem with the 4.0 installers we sent with Office which we were able to correct for all the other installers we released with 4.0. To avoid having weird multiple versions we left the original installers alone. Here are the details of the installer differences between build 58 (Full FAT, and OE PPC) and build 59 (the rest of the installers):

These are the changes that will be in installer build 59:

- The "Recommended" installers (and the "Recommended" option of the active setup installer) will no longer include "System Folder:Extensions:MS Library Folder:Classes:extra.converters.zip". In the Active Setup installer, this 3.4meg file has also been moved to a seperate download, so you shouldn't need to download it for a recommended install.
- The dates on the "Classes:" and "Lib:" folders should now match the dates on the contents of those folders.
- The IE 3.0a libs "MS IE ActiveX Lib (PPC)" and "MS Internet Library (PPC)" will now be moved from the System Folder to the trash whenever the new PPC versions of those libraries are installed. Previously, the installer attempted to delete them, which would fail if they were busy.
- The installer will now attempt to shut down all other applications before installing files, so that none of the libraries should be busy.
- On an uninstall, the installer will no longer try to delete any of the contents of the "My Personal Web Site" folder.
- The Full Fat installer is now BinHex'd, rather than MacBinary'd, since StuffIt Expander can't unpack a 17meg MacBinary file. (You'll have to look in the build 58 folder to find it, though, since it's not being rebuilt with this build.)
- The Active Setup installer is now BinHex'd, rather than MacBinary'd, so it makes it slightly more reliable, and the space increase is tiny (since the installer itself is tiny).
- Installer bug 195 is fixed. No more "MS Preference Panels.1". See the bug for details - it's complicated.

IE 4.0a

What's different, what's not?

- *We are not releasing a 128bit update with this release - it is not needed.*
- *We are not changing the stand alone JAVA installers.*
- *We are not changing the Personal Web Server installer.*
- *Base IE does have some changes - Hence, Active Setup, and all 8 of the preconfigured installers are changing.*
- *Base OE does have some changes - Hence, all 2 of the OE installers are changing.*

What are the version numbers?

- *We are calling this release the IE 4.0a release.*
- *For OE, this is actually the OE 4.0c release. We are skipping 4.0b due to the "is that a beta?" questions that we always get. The mac community is not yet used to our number convention.*
- *For JAVA and PWS, these remain 4.0 installers.*

Here is the summary of IE 4.0a bug fixes:

1) 10288 IE4.0a - ISL w/data fragment: IE or OE may crash on launch after clean install
2) 10298 IE4.0a - ISL w/data fragment: Being on SSL page in IE causes OE crash on automatic mail update.
3) 10301 IE4.0a - ISL w/data fragment: Second client launched can't use the cache.

4) 10313 IE4.0a - after entity à spaces are ignored
5) 10314 IE4.0a - ISL w/data fragment: 68k crash on launch
6) 10315 IE4.0a - Office: Crash-launch IE 4 from web toolbar, quit app. Power PC illegal instr. & unmapped.
7) 10344 IE4.0a - WEB: No navigation over Run From CD
8) 10352 IE4.0a - WORD: If Assistant is hidden, can't navigate to docs over http:// on servers requiring user authentication (password) changed the ISL dialogs to center on the main screen.
9) 10413 IE4.0a - Favorites lib: Crash on exit of second client.
10) 10411 IE4.0a - WEB: Crash occurs if URL moniker is instantiated without app calling OpenUrlMoniker/CloseUrlMoniker
11) 10415 IE4.0a - Fav lib: Library installed with installer build 80 causes 68k to crash on launch.
12) 10422 IE4.0a - User will have two versions of MS Component Lib when install IE4 over OS 8.1
13) 10425 IE4.0a - 68K only: IE crashes adding Channels
14) 10428 IE4.0a - DBCS: Using high ASCII characters in DHTML strings cause chars to be re-encoded
15) 10436 IE4.0a - WEB: crash in Favorites updating favorites
16) 10439 IE4.0a - userAgent string returns "4.0a" instead of "4.0"

Here are the details of IE 4.0a bug fixes:

- The ISL library fixes to prevent crashes when launching IE/OE/Office:
  - 10288 ISL w/data fragment: IE or OE may crash on launch after clean install
  - 10298 ISL w/data fragment: Being on SSL page in IE causes OE crash on automatic mail update.
  - 10301 ISL w/data fragment: Second client launched can't use the cache.
  - 10314 ISL w/data fragment: 68k crash on launch
  - 10315 Office: Crash-launch IE 4 from web toolbar, quit app. Power PC illegal instr. & unmapped.

- The Favorite library fixes to prevent crashes when launching IE/Office:
  - 10413 Favorites lib: Crash on exit of second client
  - 10415 Fav lib: Library installed with installer build 80 causes 68k to crash on launch.
  - 10425 68K only: IE crashes adding Channels
  - 10436 WEB: crash in Favorites updating favorites

- Changed the ISL dialogs to center on the main screen.
  - 10352 Office: If Assistant is hidden, can't navigate to docs over http:// on servers requiring user authentication (password).

- Fixed problem with concatenating words when the last character of a word is an agrave (a with accent).
  - 10313 After entity à spaces are ignored.
  - In addition to the above bug fix, we introduced support for JScript in non-Roman scripts because of bug #10313. The JScript library now supports parsing, searching, and string manipulations of scripts that contain non-Roman characters. The browser facilitates this by providing information about a page's text encoding to the JScript library. This is a Feature not a bug.

- Fixed the Dynamic html crash.
  - A very localized fix to eliminate crashes that appear on pages that set outterText or outterHTML via JScript.
    Dan is investigating this issue and will create a bug report aginst it asap.
- DBCS: Using high ASCII characters in DHTML strings cause chars to be re-encoded.
  - 10428 Fixed in element.cp

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

- The Container library bug fixes:
  - 10344 WEB: No navigation over Run From CD. The Temporary Items folder created at the root of the shared (read-only) volume issue.
  - 10411 WEB: Crash occurs if URL moniker is instantiated without app calling OpenUrlMoniker /CloseUrlMoniker.

- Installer bug fixes:
  - 10422 User will have two versions of MS Component Lib when install IE4 over OS 8.1.
  - 10439 IE4.0a - userAgent string returns "4.0a" instead of "4.0"

IE 4.01

Below is the plan for IE 4.01. One important note is that Outlook Express 4.01 will most likely not be ready to ship by 3-27 when IE 4.01 will be done. Rather than stagger releases and possibly have even more versions of the libraries, we will be holding the IE bits until OE is done. This presents somewhat of a logisitical issue as the new mac team should be taking over at this time, and anyone transitioning off the team will have nothing to do for a period of days and possibly weeks. The proposal is that the current IE team signoff on the 4.01 bits as ready to ship on 3-27. Then the new team takes responsibility for shipping the 4.01 release and syncing and testing with OE 4.01.

Mission

Produce an increasingly stable and high performing version of IE 4.0 that will survive in the market for 9-12 months.

Objectives

- significantly improve stability
- significantly improve performance (but not at the expense of stability ;)
- add support for Apple ColorSync with JPEGs (not doing PNG for this release)

Key pressures for this release

- need to deliver this release to be part of Allegro - external
- Apple still hasn't released final version of ColorSync - external
- need to get resources working on IE5

Schedule

- 1/30/98 - check-in performance work - **DONE**
- 2/15/98 - ui freeze, review progress, reassess dates - **DONE**
- 2/25/98 - feature complete, 50 bugs in RAID - **DONE**
- 3/4/98 - 25 bugs in RAID
- 3/11/98 - zero bugs
- 3/27/98 - IE RTW

* Please note: we are going to manage RAID aggressively to meet the bug counts listed above on those dates. Additionally, we will be constantly reviewing every bug in the database to measure it against our objectives, and its potential impact on stability.

Feature Work

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

ColorSync

We now support embedded ICC Profiles in JPEG images. We do not support them in GIFs, and we still do not support the PNG format. There are prefs located in the Web Content Pane of the IE Prefs window. Specifically the following checkboxes are available:

- Enable ColorSync - this checkbox defaults to Off. When it is dimmed, the other two are dimmed as well
- Screen - when enabled this renders any images with profiles to screen using their profile
- Printer - when enabled this renders any images with profiles to the printer using their profile

(Images with no profiles will be rendered as they always have been regardless of the state of the checkboxes. Changes in these checkboxes won't take effect until the page is reloaded.)

We are using ColorSync 2.1.2 which is the latest version and shipped with MacOS 8.0. ColorSync 2.5 is shipping soon from Apple, and it wouldn't hurt to validate against this version. In theory there shouldn't be any problems working with prior versions starting with 2.0 but dev has not tested against these. ColorSync 1.0 should not work.

(FYI Test: ColorSync 2.5 will ship with an AppleScript that embeds profiles.)

UI

- **Back and forward items have been added to the contextual menu.** Now when users click and hold or control click within a browser window, they will be able to access the forward and back commands from the contextual menu.
- **Ability to page down using the space bar** (option+space bar to page up). When in the browser window, users press the space bar once to move down a page at a time; option+space bar to move up.
- **Contextual menus have been rearranged** to put more popular features toward the top--should check for consistency in arrangement.
- **Fixed frame printing.** Previously, when the user had no frame selected on a page with frames, the print command would be disabled. Now, IE will print the largest frame on the page when no frame is selected.
- **Favorites usability change.** If the Favorites window is in front, and a browser window is open with a web page loaded, you will now be able to use the "Add Page to Favorites" item from the Favorites menu. Previously, this menu item was disabled when the user was in the situation described. No other scenarios or menu items are affected by this change.

ICW Changes

Apple has asked that we redo the user interface of the ICW for our release as part of the Allegro bundle. We are planning on having these changes ready, tested, and shippable simultaneously with IE/OE 4.01. However, since Apple's Allegro schedule is still in flux we are going to hold the bits as our current ICW functions fine, and if Apple wants changes after we ship 4.01, we won't have yet another version.

Performance Work

We focused on several areas:

- Table formatting
- less redundant drawing
- start loading images earlier
- improved internal threading model
- update the status bar a lot fewer times (not a visible change to the user)

- ISL is slightly less conservative about validating image expiration on the net (reduces network traffic when reloading)
- rearchitected the internal idle mechanism to not spend time idleing objects that don't exist or don't need time

Most importantly we have added support for timing the loading of documents to enable us to have some consistent performance metrics. This functionality can be turned on by adding a PeRf resource with an ID of 0. We only have 4.01 builds with this capability. Dev feels pretty good about this being a very important way for test to monitor our perf across builds.

Installer Reduction

We are reducing the number of installers for 4.01. Active Setup is getting more stable, and we have some good data on what was popular and what wasn't. We had some discussion where alternate suggestions were made. Specifically, we need to continue to do a Full Fat installer for IEAK and other things. Another suggestion was to replace the two recommended with the Fat alone. Unfortunately, the customer download numbers (which i have added after some of the installers) and relative sizes don't seem to support this recommendation. This means that as of now we have only killed 3 installers. However, if the Minimum Fat installer is only marginally bigger than the Minimum PPC I think we should replace the two minimums with the one Minimum Fat. Let's discuss further and get this one settled.

- Active Setup - 4.01
- Full Fat - 4.01 - 169
- Full PPC - 4.01 - 488
- Full 68K - 4.01
- **Recommended Fat - KILL - 29 (and 3.4 MB larger than PPC alone)**
- Recommended PPC - 4.01 - 468
- Recommended 68K - 4.01
- Minimum PPC - 4.01
- Minimum 68K - 4.01
- **Java PPC - KILL**
- **Java 68K - KILL**

- OE PPC - 4.01
- OE 68K - 4.01
- PWS Fat - already done
- 128-Bit Fat - 4.01

doScript AppleEvent

For a variety of reasons including automated testing we will be adding support for a "doScript" AppleEvent in IE. The AE will resemble the one for HyperCard and Frontier, and will act on the frontmost window in the browser.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Bundle Apps

NetMeeting

Blake Irving and John Scarrow own NetMeeting. They currently have no plans for a Macintosh version. However, many months ago they made a deal with Netopia (nee Farallon) to work on a Macintosh version. Specifically, Netopia was going to produce a mac application that did the application sharing functionality that NetMeeting provides.

At MacWorld San Francisco, we spoke with Forest Milkowski who is a product manager at White Pine Software and a friend of DonBrad's. We talked with Forest at length about him producing a netmeeting compatible client that does the audio, video, and chat components of the product that presumable Farallon is not doing. Forest was willing to change the CU-SEE-ME ui to match that of IE, and even call it something like Net Meeting for Macintosh – powered by CU-SEE-ME.

With changes in priorities, and not being sure about the long term status of the mac internet products, I have not pursued this any further. In the meantime, Blake Irving has told me that he thinks White Pine and Netopia are now collaborating on some offering for the Mac. I haven't contacted Forest or the guys at Netopia, but I think somebody should to follow up.

NetShow

Richard Saunders owned the NetShow 2.0 product for the Mac. I think Ramiro Calvo owns it for the 3.0 timeframe. In fact, Ramiro and a bunch of guys from the company contracted to do the next mac version came by and talked to me and BowenS. They are interested in leveraging off of our code (ISL, Favorites, Prefs) and also are willing to modify their UI to look more like ours.

Additionally they want to add support for QuickTime streaming to their app. This is a hot button issue for Apple as they view NetShow as major competitors for QT. Not quite sure what the scoop is here, but Ramiro had a meeting with Apple, and warned me that Tim Schaff from Apple would be sending me e-mail to ask me why we wanted this in NetShow. I'll be kicking this up to the execs and seeing what they want us to say before I respond.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Win32 Technologies

Brief List of Differences between MacIE4 and Win32IE4

- Multimedia controls
- VB Script
- SMIME
- CSS frosting
- MHTML
- Accessibility
- LiveConnect
- PFX
- OSD
- Wallet
- Editing
- Kiosk mode
- personal certs
- Active Desktop
- history by site
- server gated cryptography
- Save as desktop pattern
- enumerate intranet sites
- *(also see UI section for comprehensive list of UI differences)*

VBScript

Regardless of the different opinions as to whether we should or shouldn't support VBS as a company, for better or worse we've done way too good a job evangelizing it and using it on our own site. Unfortunately it is not so simple to support. Andrew Clinic is the PM on the scripting team in Redmond. They view it as their responsibility to help us get VBS in IE for Mac. If we push they will most likely get resources working on porting the engine. Unfortunately there are major IE issues that have to be resolved (case sensitivity, how certain variables are handled) to make this work. It is not the end of the world if it doesn't, but it probably should happen for IE5.

XML

We met with Chris Lovett of the XML team. They are writing entirely new engines for IE5 for XML and XSL. They based their work on the Java version they had for IE4, but they are rewriting in C++. They view it as part of their mission to codein a cross-platform sensitive fashion as well as to deliver their engines cross-platform. However, they are also viewing as win32 their #1 priority and the mac as their 4th priority after win16 and unix. Given the SimShip assumption we were making for IE5, and them not sim-delivering their components, I made the call that we would not do this for IE5. Additionally, as other than CDF (which we support in a non-robust fashion), and OSD (which we don't support and aren't planning to support), there are no applications for the XML stuff and I don't believe there will be any major ones until after IE5 ships.

HTML Help

I spoke with Chad a few months back and there is a team working on HTML help for win32 up in Redmond -- Kate Harper and Ralph Walden. That team had considered hiring a contractor to port HTML help to the mac. Thinking that they would want to use our IE control as the basis I contacted them a couple

of times but they never responded and I gave up. I'm assuming they aren't actively pursuing HTML help for the mac. Apple however is pursuing it, and that is covered below in the Apple section.

Trident

It has been clear for some time, and has become even clearer in the IE5 planning process that given the current course of IE for Mac, it will never have feature parity with the trident engine. Furthermore, the gap is going to only get worse as Trident moves forward at full speed and the fundamental architecture of the Mac browser is incapable of supporting the features going into the win32 browser. Porting trident has been one proposed answer to this problem. For a time, because of trident's working set issues, this seemed non-viable. However, I think that the trident team has made some fantastic strides in the areas of footprint and performance, and at the very least if (and this is a big "if") we do want to get to feature and rendering parity between win32 and mac IE, we should spend some time investigating porting trident. I would suggest that the only path to success for that option is for the win32 trident team to be heavily invested in making it work (even dedicating resources).

DirectX Multimedia Controls

EricEng's group has hired a team of mac contract developers to port DirectX3 to the macintosh as well as the multimedia controls that are based on directx and run in the win32 browser. Due to lack of testing time, we did not ship them with the 4.0 browser, and they were outside the scope of the 4.01 browser. Pablo Fernicola has managed the relationship with the contract team. The UNIX and Win16 teams will never have this functionality in their browsers. The new team will need to decide whether these are worthwhile. My personal opinion is they are probably worthless without vbscript support, and even then I'm not sure who is ever going to really use these and expect to run anywhere besides win32.

CSS Transitions and Filters

Jay Jacobs has done the initial work to make the CSS transitions and filters work in the mac browser. However we did not have time in the 4.x timeframe to include this functionality in the browser. The code is still available. The win16 and unix browsers will never support these. These are more valuable than the multimedia controls, but again questionable in my opinion.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Java

Our strategy for our Java efforts is coming to a head. However, here is a brief list of things the Mac VM does and does not support:

Does support:
- JDK 1.1.4
- CAB decompression (in addition to JAR)
- AFC
- JManager 1.0

Does not support
- Package Manager
- Authenticode
- RMI
- JNI
- JRI
- J/Direct
- JManager 2.0
- JDBC
- DirectX/J
- MS debugging apis
- Scripting of Java (COM in the VM)

DCOM in Java

There are currently two methods for doing object to object communication in Java. Sun is pushing RMI and Microsoft relies on DCOM. However, there is no DCOM for the Mac, and for strategic reasons we are not supporting RMI on the mac, which unfortunately leaves our users/developers with no options in these areas. Nat Brown's team was supposedly doing a version of DCOM in Java that we could ship. Last time I checked they were not making speedy enough progress for us to include them with our 4.0 releases.

Current Strategy

Get Apple invested in our APIs so we don't have to continue investment in a mac java implementation. Specifically: they will add our security apis, debugging apis, and support for J/Direct, at which time we will drop our VM and support MRJ exclusively. Until then we use our own VM including in the operating system bundle.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Marketing

We'll go over this verbally.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Apple

Master Agreement

This kind of lost steam. Rick Holzli had at one time promised a master agreement that would relieve us from individually licensing bits and pieces of their OS to ship with our product. Chad was going to move on this and I don't know where it is at but it needs to be resolved.

IE Control

Apple has implemented a version of AppleGuide that uses the IE version 3 control. However we have never granted them a license to use it. My personal opinion is that they should not receive a license to use this as part of their OS until we are the exclusive browser in the OS, Netscape is not on the CD, and we are oput of the cold peace – i.e. steve jobs is actively promoting our browser over the competition.

Cyberdog Speed improvements

Ed Tecot brought up some speed improvements he thinks we should license from Apple that are present in Cyberdog. We did not spend any time considering them for the 4.01 release. They should be considered first on a technical basis and then on a business basis for a future release.

Scripting Engines

Andrew Clinic owns this and is prepared to put a resource (maybe) on porting the vbscript engine to the mac. However there is a ton of work required on the IE side to get this in. Andrew also owns the licensing stuff with Apple over the scripting engines. Chris Espinosa at Apple is his contact.

Allegro

This is TBD delivered by Lee this afternoon. However we should designate someone to take over from Lee from the new team asap.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Key Contacts


Ross Ely.msg


Mike Dodd.msg


Charles Wiltgen.msg


Tim Schaaff.msg

Ross is the marketing guy at apple for QT. Mike Dodd is a key engineer we have been dealing with. Charles is the evangelist and Tim runs the group.


Craig Eisler.msg


David Petchey.msg


David Vasquez.msg


Pablo Fernicola.msg

Craig is Pablos boss and initiated the mac effort and managed it for awhile as far as directx on the mac. David Petchey was hired to write the Java libraries to DirectX (although they are in Java they are laced heavily with native win32 code and will not easily work on the mac). David Vasques works for Post Digital software and is porting DX3 and the controls to the mac. Pablo Fernicola works for Greg and manages the relationship with Vasquez.


Andrew Clinick.msg

PM for the scripting engines. Also managing stuff with Apple on licensing.


Karl Jacob.msg

Runs the directX team. We are considering bundling their liquid motion Java libraries with our VM.


Gary B Little.msg


Peri Frantz.msg

Gary is the Marketing guy for MRJ, Peri is the dev manager.


Dave Fester.msg

Our marketing guy in IE marketing.


Jonathan Kahn.msg

Our contact at Aladdin is Jon Kahn. The deal we made with them has two components. We paid them to port the CAB engine that we use for Java – this is a one time thing. We also, paid them a license fee (which is annual) for using the stuffit engine. Verbally Jon and I also agreed to give them an active channel at launch (which we did), space in our booth (which we did), and access to our OE apis so they can add OE to their Magic Menu product. In exchange Aladdin was going to give us 3 more formats in addition to the 2

we already have licensed from them. This agreement comes around in May and should be reexamined and possibly cancelled.


Rick Holzli.msg


Shaan Pruden.msg


Jordan J Dea-Mattson.msg

Rick is of course our evangelist. Shaan was our Java evangelist but since the reorg Jordan is now our Java guy.


Rhonda Middleton.msg


David Farrington.msg


Mimi Kennedy.msg

Rhonda works for Mimi in Austin at licensing. They made licensing TEC a miserable experience. David Farrington helped us out making an exception for MS in how we licensed.


Suzanne Sylliaasen.msg

Suzanne is our MTS contact for IE.


Mike Reed.msg

Mike Reed runs a company in North Carolina of a bunch of former QDGX developers who have come up with a lightweight replacement/addition to QuickDraw that is multi-threaded/re-entrant, and fast. We thought about licensing this for our Java efforts, but decided we didn't have time to take advantage of their stuff.


Scott Kovatch.msg

Scott worked on the metrowerks VM for us.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc


Maggie Waggoner.msg


Erich Andersen.msg

Maggie is our legal contact for IE (as is Cory Van Arsdale). Erich is our contact for Java.

Redmond Key Contacts

Charles Fitzgerald – GPM for Java
Chris Jones – owns IE
Hadi Partovi – GPM for IE
Joe Peterson – dev manager for IE
Eric Berman – IE GPM for UI
Scott Berkun – PM works for Eric
Castedo Ellerman – PM works for Eric
Julian Jiggins – dev lead for IE UI
Eric Henning – PM on IE for services (ISL) works for Hadi
Ray Sun – PM on IE works with Eric Henning (deals with MHTML, client capabilities, and stuff like that)
Tom Yaryan – Lead PM for TCO stuff
Tim Johnson – PM for IEAK
J.K. Bouwens – PM for3rd party add-ins (realaudio, etc.)
Christian Fortini – runs trident
Michael Wallent – GPM for trident
Rod Chavez – runs dev on trident
Arye Gittleman – dev lead for trident (helpful with tables and netscape compat)
Alex Avram Shtaygrud – did netscape compat work for trident
Sara Williams – trident PM works for Michael (I think she is doing CSS)
Laurie Anna Edlund – trident PM works for Michael (was doing CSS – office9 properties)
Laurent Vernhes – runs test for trident
Ramesh Parameswaran – runs UNIX IE
Lotfi Herzi – runs win16 IE
Ramiro Calvo – netshow mac contact

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Admin

Mac IE Mailing Lists

Mac IE Team – all internal communications
Mac IE Status – all external communications

I will transfer ownership of these DLs after 4.01.

Mac IE 4.0 and IE Mac Status are owned by Steve and Don respectively and should be killed.

Other Mailing Lists of interest

- AICG Webmasters
- Apple NDA Notification
- Cross-Platform QFE Issues
- IE Core Team – Browser & Shell
- IE End User Experience Team
- IE Leads – Browser & Shell
- IE PM Leads – Browser & Shell
- IE PMs – Browser & Shell
- IE Web Computing Team
- IE5 Browser PMs
- IEAK (Mac) Customer Feedback
- Internet Client and Developer News
- Internet Explorer Feedback
- Java Class Lib Team
- Java Mktg Swat Team
- Mac IE Build
- Mac IE Library Build
- MAC IEAK
- Mac Java VM Deltas
- Mac Java VM Strike Team
- Mac Program Managers
- Macintosh Internet Explorer External Beta

Web Server

I did an overhaul of the MS-Bay internal web site. The data there was updated through the waning days of IE4 and hasn't been updated since. All the information is up there under the old home page. Nothing has been deleted. Nor have I managed to convert a lot of the old still relevant content. It seemed like a waste considering the reorg.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Misc

128 Bit Security Testing

We need to get a test account with a 128 bit secure site so we can test our 128-bit patch. Wells Fargo won't give us a fake account so somebody should take $100 and open one up.

User Agent Strings

- Mozilla/3.0 (compatible; MSIE 4.0p2; Mac_PowerPC)
- Mozilla/3.0 (compatible; MSIE 4.0p2; Mac_68000)
- Mozilla/4.0 (compatible; MSIE 4.0; Mac_PowerPC)
- Mozilla/4.0 (compatible; MSIE 4.0; Mac_68000)

CSS Follow-up

This is a good place to get status on our CSS. Last time I checked he had yet to update it, but we should keep checking (he was testing against preview 1).

http://www.mcp.com/hayden/internet/style/table.html

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Channels and channel guide

Currently, there are two separate channel guides housed on the MS channel guide server: the "optimized for Macintosh" channel guide, and the regular Win32 IE4 channel guide. The Win16 and Unix 4.0 browsers do not currently have channel partners. The Macintosh channel guide resides at: http://iechannelguide.com/guide/en/en_us.asp?mac=1; this link is hardcoded into Mac IE. Although the Win32 guide has different categories of channels, the Mac channel guide simply has a list of channels (19 at last count) on the Mac. The channel logos are displayed 7 at a time, and channels rotate, so that there is no system as to which channels are listed "above the fold." The only distinction between "Gold" channels (channel partners who signed a contract) and "basic" channels (no contract) are that the basic channels are listed in text format in the guide versus having color logos.

Technical contact: The contact for the channel guide at the inception of the project was Mary Haggard mailto:maryha, but currently I am dealing with Cilla Marriott mailto:cillam. This is for technical channel guide issues only. Whoever takes this aspect of the project over should contact Cilla for a username and password for the channel guide database, as the current set of Mac channels will have to be monitored.

Channel contracts/development: Business development contacts (channel contracts, etc.) are Jennifer Cooper mailto:jcooper and Michelle Davis mailto:micdavis , who work in the Foster city office. They own the contractual/marketing relationships for the following channel partners, who all originally constructed Win32 channels and "ported" them over to the Mac channel guide.

- Bloomberg
- Wired
- Hollywood online
- Pathfinder (includes Time, ParentTime, Dr. Weil, Fortune, and People channels)
- Discovery online
- New York Times
- C|net/Mediadome
- HBO (HBO has not signed a contract or produced a Mac channel yet--we are currently in negotiations with them)

I am the technical contact for all the above channels, meaning that I give them early builds of the browser, work with them to tweak their content, etc. Some of these channels are farther along than others (Wired and Discovery versus Bloomberg, for example).

Here is the technical contact information for the above channels:

I am the contractual/marketing contact and the technical contact for all the "pure Mac" partners, who are the following:

- Macworld/Macweek: Jeff Julian (contractual contact), Roxanne Gentile (tech contact)
- Aladdin Systems: They are not a gold channel, so they did not sign a contract, but the contact there is Jon Kahn
- Macaddict: Cheryl England (mailto:cengland@macaddict.com) is the contractual contact. Mark Simmons (info above) is the technical contact.
- TidBITS: Adam Engst (info above), contractual and technical contact

- MacHomeJournal/NetProfessional: Raines Cohen (info above) is the technical and contractual contact.
- Apple (this is a special case, as described in "open issues."): Greg Gilman is the technical contact; Rick Holzli (our MS evangelist, mailto:holzli@apple.com) is the contractual contact.


Raines Cohen


Roxanne Gentile


Greg Gilman


Adam Engst


Mark Simmons


Jonathan Kahn.msg

Currently prepopulated channels/web pages (i.e., these show up in the channel bar at first install): There are currently three items that are prepopulated in the IE4 for Mac channel bar:

- **Introduction to channels** (a web page describing what channels are and how to use them in IE4); this was developed by ctrood at MSBay and is hardcoded into the product. Due to some major UI changes that the Win32 team is making (see the UI section for more details), this will have to be revved extensively if we follow suit.
- **Internet Explorer for Macintosh channel**: This channel is maintained by the IE web team and is currently housed on the ms.com server. Contact Molly Dempsey mailto:a-mollyd and/or Emily Warn mailto:emilyw about this channel. This channel is presubscribed, and has been localized by the localization team (see James Carroll for more details on channel localization).
- **Active Channel guide:** This is the link described above.

In the Mac version of the browser, the channel bar logos do not appear unless the user is connected to the Internet. The logos for the Introduction to channels and active channel guide pages are housed on the channel guide server. The logo for the Internet Explorer for Macintosh channel is housed on ms.com.

Localization issues: There are currently no Mac channels available in other languages except for the Internet Explorer for Mac channel. This is largely due to the fact that the international evangelism teams were not given enough time to go out and evangelize Mac partners, and not many of the original channels were easily transferrable to the Mac browser due to lack of support for VBScript, filters, and transitions. I will discuss the future of the channel guide in the "recommendations" section, but the way things stand now is that if the international DRG drums up enough international support for Mac channels, we should address the following issue: namely, how do international users get presented with channels in their own language and locale? Mac IE can only sniff for language (e.g., Spanish, English), not locale (e.g, Argentina, Australia); this is apparently a platform issue, so we will have to include a mechanism in the channel guide through which users can select the locale of their choice. This remains an open issue with the channel guide team that will only be raised if we end up getting a number of people who want to develop international channels.

Open issues:

- **The Apple channel:** Apple is included in the Mac channel guide due to the coverage that our overall comarketing agreement with them supplies. However, they have refused to comply with some of the terms that are specific to channel partners, namely, promoting the active channel on their web site (i.e., putting an "add active channel" button on their web site that would link to the Apple channel), and putting a link to download Mac Internet Explorer 4.0 on their web page. The business development team has told us that unless Apple complies with those two requests, we should pull the channel; this situation has to be managed very carefully, as we do not want to alienate Apple, but there are some basic comarketing that they should be doing with us as the default browser.
- **Separate channel guides:** Having separate Mac and Win32 channel guides is not an optimal situation, especially since our team has traditionally not had the bandwidth to single-handedly evangelize mac-specific channels. The reason we have two separate guides is because most of the channels that were originally created on Win32 use VBScript, filters, and transitions, which are not supported on the Mac. Also, the Win16 and Unix teams did not release their 4.0 versions with channels. Moving forward,

there either needs to be full evangelism of a strictly Mac channel guide (e.g., actively recruit more channels for the Mac guide) or some sort of rationalization between the two guides. Currently, only one person has been recruiting and providing technical information for channel partners; this cannot be sustained long-term: there needs to be constant communication with and support from DRG if the channel guide is to continue.

Cross-platform authoring documentation:

- **Crossplatform authoring guide.** Sjoert Ebben on the international team has developed a crossplatform channel/DHTML authoring guide. This guide was designed to be used for the international groups to help evangelize crossplatform channel development, but would definitely be of use to the US teams as well. This document has yet to have a full, detailed technical review by our team. The Mac IE team needs to engage with Sjoert and make sure this document is technically accurate as we move forward with IE5.
- **InetSDK.** Jerry Drain is the contact for the InetSDK. This is an extremely valuable and informational document that is heavily Win32-centric and does not have a lot of information about how to write crossplatform IE4 content. The SDK teams were asked to look more closely at crossplatform issues for the IE5 timeframe. They are currently in the middle of planning the next version, so we should have a UE representative engage with Jerry's team to make sure there is time to contribute and review.

Recommendations for future development of channels in IE5:

There are a few things I think are essential to move forward and maintain a credible channel (and credible crossplatform) story on Mac IE:

- **Merge the Windows and Mac channel guides.** This is not as simple as it sounds, since it means that the Mac IE5 team will have to sign off on some baseline compatibility to move forward, namely:
  - **IE4 DHTML support:** This means fully implementing CSS positioning, and adding filters and transitions.
  - **VBScript support:** Regardless of the lack of adoption on the Internet, the primary scripting language used for all web content at MS is VBScript. Also, 60% of our channel partners use VBScript. The DRG group heavily evangelizes DHTML written with VBScript. We could spend time trying to educate the rest of MS that they should be writing crossplatform DHTML written in Javascript, but this has proven quite difficult in the past and has really hurt our crossplatform story. VBScript support is essential to improving our channel and DHTML stories on the Macintosh.
- **Engage with Will Poole's business development team and get them signed up on the crossplatform story.** The current plan for IE5 Win32 is to get rid of the channel bar, meaning that channel partners that originally got premium space by being prepopulated in the channel bar will no longer have that opportunity. This turns into an opportunity for the Mac team because the Business Development team will have to reestablish comarketing agreements for the channel partners. The Mac IE team needs to be part of the discussions surrounding new comarketing relationships and get on the bandwagon that was missed the last time around.
- **Engage with DRG and the crossplatform authoring teams to ensure that the appropriate technologies are being evangelized as "crossplatform."**
- **Ensure that there are appropriate resources signed up to work with channel partners and the channel guide team.**

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

User Interface status

For the past couple of months, I have been meeting with the Win32 IE UI team to work towards UI consistency (where it makes sense) between the Win32 and Mac products and also ensure that the Mac team is involved in discussions regarding new IE5 functionality. In my first meeting with Eric Berman mailto:ericbe , the group program manager in charge of the IE5 UI team, we cataloged and developed a spreadsheet of UI differences between our products, with recommendations as to which items the Win32 team should change, which items the Mac team should change, and which items we should leave alone for now. I then met with John Stephen to go over the recommendations and evaluate them according to our resource assumptions at that time. I've updated the spreadsheet, which resides at http://ie/specs/secure/Mac-Win32Diffs.xls on the ie5 specs site on http://ie, for all the items for which I am labeled as owner. At this point, whoever takes over this aspect of Mac IE needs to reevaluate these recommendations based on the current resource situation and product goals, and also follow up with the rest of the owners listed on the spreadsheet. Scott Berkun (mailto:scottbe), who reports to Eric, is a good general point of contact for these issues (see the Win32 UI team contacts section of this document for further information.)

Consistency changes

Although these issues are listed in the spreadsheet, here is a summary of the consistency changes that were agreed upon by John Stephen and myself while doing the initial planning for IE5 (:

Definite (we know we want to do this, if time permits)

- resize explorer bar pane
- add option to save history by number of days as well as by number of items
- sort history by site as well as date
- under view menu, put button bar (but change to "standard buttons"), Favorites bar, etc., under hierarchical "toolbar" menu; add "text labels" to this as well
- change "offline browsing" to "browse offline" in menu and dialogs
- have IE check whether or not it is the default browser on startup (with "don't ask again" option)
- print background colors and images
- add print to context menu
- in proxy prefs, change "enable/disable" to "use a proxy..."
- add "clear history" to prefs

Probably should

- Move Favorites import/export commands under Favorites menu (space may be a concern, but we do have the Favorites bar)--this is a usability change that stemmed from some of our site visits.
- put "outlook express" and/or other integrated apps under the go menu (space issue is same as above)
- Change the name of the network preference to the "connection" preference
- Add large icon option to toolbar
- option to print linked documents (similar to save as... options) and make frame printing option more prominent
- for edit/find, add Command-shift-g to search up and down?
- empty cache when quitting browser option

Needs a lot more thought

- possibly doing a "view cache" option, maybe in the cache prefs?
- send page (this depends on mhtml support)
- http errors and timeouts: should look at win32 spec and see what we think (up on http://ie/specs)
- have to ask bowen about auto-domain scanning: .com, .edu., .org in autocomplete, also possibly fixing common syntax errors
- possibly using OE multi-user panel in IE
- get rid of the search button? Search pane will be fully customizable but will not work well with all content. However, users can also use a favorite to get to their favorite search page if they don't like how it shows up in the pane. We will try to address this issue during the usability tests.

Major new Win32 UI changes/features:

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Although most of this can be found up on http://ic/specs, here is a summary of the major UI changes that are happening on the Win32 side that we should be aware of:

- **Channel bar is going away.** This UI change has already been implemented in IE5 Win32. Channels are now displayed in the Favorites bar, and developers can create custom icons as part of the CDF spec instead of having large custom logos. Since Mac IE has implemented channels like this in the first place, it should not be too much trouble to yank the channel bar from the UI, however, Scott Berkun is working to get business development signed off on this plan, and we should confirm with him before going ahead and doing it. Also, the Mac IE team needs to ensure that whatever plan busdev comes up with for dealing with ICPs includes a solution for crossplatform content.
- **Organization of Favorites, subscriptions, and offline items are being improved.** Currently, the Win32 IE team is testing a variety of prototypes for organize favorites. As with the Mac product, they are hoping to include the ability to view and organize favorites, subscriptions, and offline content all in the same area. Castedo Ellerman mailto:castedoe and Scott Berkun are in charge of the offline and favorites specs, respectively, and these specs are all up on the web site. Most of the changes they are thinking of making are actually making themselves more in line with the Mac product, so there should not be too much work on our end to implement this.
- **Http error spec.** Win32 IE is going to provide users with more intelligent error messages, and have created specs and templates that we should be involved in developing and eventually use in IE5.

Non Win32 UI changes:

In addition to driving consistency and involving myself in crossplatform UI improvement discussions, John Stephen and I were also tasked with coming up with three "big" features that would help make Mac IE worth a full version number jump to users. (Although adding DHTML support is incredibly difficult and takes a lot of resources, unfortunately that type of work is not incredibly visible to most users.) The following specs were given to the Win32 IE team, some of which are being considered for the IE5 release (e.g., customizable toolbar) and some of which will probably wait until Win32 IE6. The Mac IE team been given the go ahead to execute on these specs if time and resources permit. I have mentioned other cross-product issues involved (e.g., with OE and Office) later in this document. Included at the end of this document are the specs for Customizable toolbar, Download explorer bar, and Mail notification from within the browser.

UI changes that were completed for 4.01:

Only a few very small UI changes were made for 4.01, so that John and team could focus on fixing bugs:

- Added back and forward to contextual menu
- Added ability to page down with space bar (page up with option and space bar)
- Favorites usability changes: If you have any Favorites window in front and you have a browser window open with a web page loaded, the "Add Page to Favorites" menu item will be enabled and will add a Favorite for that web page to the frontmost Favorites window. There are no other scenarios or menu items affected by this change.
- Rearranged contextual menus
- Fixed frame printing so that, when no frame is selected, the Print command will print the largest frame on the page (instead of being mysteriously disabled as it is today).

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

4.01 changes that were bumped to IE5:

There were also a few "no-brainer" changes that John wanted to make for 4.01 that did not make it in the 4.01 timeframe (these are all items that weren't specifically listed as bugs):

They are:

- **autocomplete enhancements** (a usability suggestion); this includes:
  - adding a magnifying glass icon to the address bar arrow to indicate that this is a type of search
  - adding a tooltip to the address bar arrow when it is greyed out to explain what the feature is
- **Favorites delete warning dialog** (hitting the delete key currently just wipes out selected Favorites, so an "Are you sure?" dialog would be added).
- **Offline link-status cursor.** This is good for us to add but should be consistent with the new windows status icons/cursors. (Jenn Shetterly would be the contact for this).
- **Drag-download images**: Instead of just dragging and downloading, as with links, users will option-drag to download images (thus eliminating the problem of having a link with an image and downloading the image instead of the link).
- **Drag from Favorites Bar** (dragging to trash to delete from Favorites Bar)
- **Drag browser selection.** Drag any selection of text from the browser and drag it to the Finder as a text clipping. (Might also be good to improve text formatting when dragging/cutting and pasting into Word documents)

Wishlist:

These are some ideas that are Win32-compat, customer, or internal suggestions, not in any particular order. These had yet to be sorted out against resource requirements. Also, these suggestions may have duplicates in the bug database, so they should be checked against that. Jsteph also has a handwritten document that may have some additional suggestions/work items.

- **Full screen mode:** Gets rid of toolbars, status bar, explorer bars, all at once and resizes to fit the screen
- **Kiosk mode:** Not just full screen, but certain features, navigation elements are locked down
- **Save as desktop picture:** Could be included on the contextual menu when control-clicking on a picture: analogous to the "set as wallpaper" feature in Win32 IE
- **Ability to add more favorites to the toolbar** (either multiple favorites bars or putting a Folder on the Favorites bar, click and hold to get favorites list)
- **Browse file system by dragging folders to browser** (drag a folder onto Netscape Mac to check out this feature)

Win32 UI team contact list:

General email alias: IEEUE
Eric Berman: Group Program Manager
Scott Berkun: Search page, Favorites/subscriptions issues, general UI issues
Jenn Shetterly: mailto:jenns : Product designer: icons, look and feel, splash screen , etc.
Gayna Williams: mailto:gaynaw : Usability issues, talk over how features evolved, etc.
Castedo Ellerman: Offline UI
Ray Sun: mailto:raysun : Save as MHTML UI

Other issues:

Private beta: The private beta group has not heard from the IE team since 4.0 shipped. I would recommend rolling the IE private beta into the Office private beta structure as soon as possible, but you should contact Christin Overton to get all the names that we used.

UI Specs

I. Customizable toolbar

summary

A customizable toolbar feature that allows users to:

- rearrange toolbar buttons on the fly
- create a single custom toolbar that has only the buttons that the user wants, arranged in the way the user wants them; user should be able to switch from this toolbar to the default toolbar(s).

design goals and justification

This feature spans a few different vision areas:

- **Ease of customization:** customizable toolbar is one of the most asked for options by both corporations and end users (cf. a recent list of common asks from the field that has been going around); it is our goal to produce an intuitive, simple way of providing this functionality.
- **Best story for laptop users:** most laptop users do not have a premium on screen space and need to be able to configure the toolbar to match their environment.
- **Simplicity:** instead of trying to anticipate the needs of a diverse set of users by loading up the toolbar, using customization, we can target individual needs and provide an efficient browsing experience for a variety of scenarios.

scenarios

Some possible scenarios:

- John Smith does a lot of web development, so he wants to a Save as button in a primary spot. However, at times when he's just surfing and not doing any web development, he would rather have the stop button in the same spot.
- Sally Johnson has a laptop, and when she uses IE on the small screen she loses the Print button, which she likes and uses a lot. She never uses the Full Screen Button, and wishes she could get rid of it and put Print in its place.
- Harry Baldwin is a corporate administrator who wants to cut down on training costs by having a toolbar that has only what he deems to be the most important functions. He doesn't want his users to be able to rearrange the buttons (IEAK scenario).

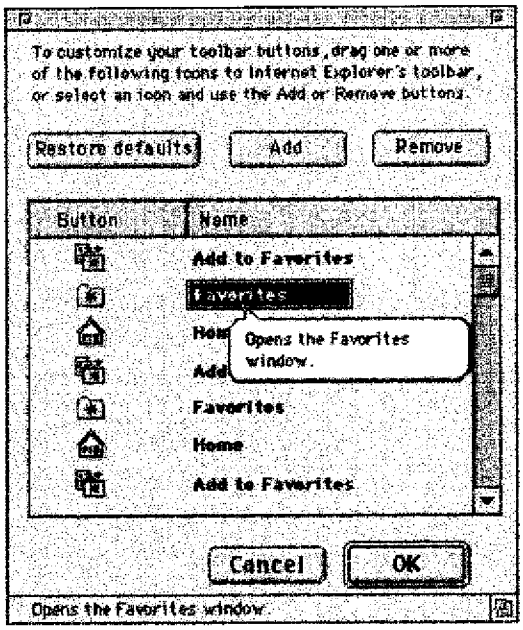
design

Users will get to the feature by either selecting a menu item (possibly under Edit for Mac), control-clicking (right-clicking on Windows) on the toolbar for a contextual menu that has an Edit toolbar option, or holding down the command key to drag toolbar buttons around.

Adding and removing buttons

Once users select Edit toolbar, a floating palette containing all the available buttons will come up. The window contains a picture of each button, the name of each button, and a description that is enclosed in a tooltip. If the user has tooltips turned off, the description will be displayed in the bottom of the window, as shown. Users have a Finder-type view of the buttons, so they will be able to click the cursor in the button name area and easily rename the buttons, just as if they were in a Finder window or in the IE Favorites

window. Users will have the ability to drag buttons from this window to the toolbar, or select a button and use the add and remove buttons.

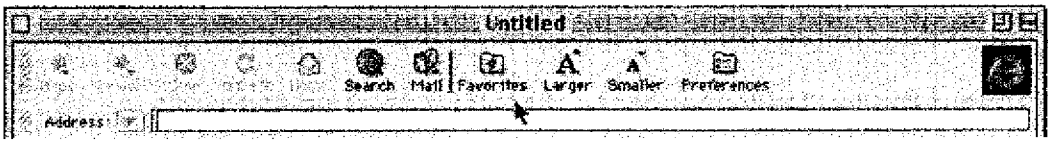


Users should be able to drag buttons from the active toolbar to the trash as well, either by holding down the command key and dragging, or clicking, holding, and dragging out of a certain area, thus corresponding to the way that dragging links works (i.e., if you click and hold on a page link and drag far enough, you can drag the link to the Finder; otherwise, the user's behavior will result in either clicking on the link or bringing up the context menu.) Buttons that have already been added to the toolbar or currently exist on the current toolbar will be greyed out/disabled.

In terms of how many buttons can fit on the toolbar, it will work like toolbar favorites, in that you can drag an unlimited amount of buttons to the toolbar, but you will only be able to see the buttons that visually "fit" in the space. Users will be able to fit more buttons by changing the text labels, as mentioned above.

Moving buttons around on the toolbar

The cursor will turn into a hand (not the cursor as displayed below, but the "grabbing" hand which is shown in IE4 when the user rearranges the toolbars themselves) when buttons are available for moving. (It's still under debate as to whether this should be all the time, or only when user clicks on Edit toolbar or uses command key to change into editing mode.) When buttons are dragged, they will manifest as a ghosted version of the icon. There will be an insertion point in the form of a thick blue line (same as when rearranging toolbars) that will help users orient the buttons (see the extremely rough sketch below).



Toolbar persistence

Editing buttons

When the user first uses the edit toolbar command to open the editing palette, a toolbar named custom will automatically be created, and all the changes made will be saved to that toolbar. The custom toolbar can then be accessed under the toolbar menus (Mac does not currently have, but will have a hierarchical toolbar menu under the view menu in IE5, just as in Win32) and through the contextual menu on the toolbar (control-click on the toolbar to access). This custom toolbar will be saved when the user clicks ok in the editing palette.

Rearranging buttons

If the user rearranges the buttons without using the edit toolbar command (by just dragging the buttons around), that button arrangement will automatically be saved to "custom" the next time the user opens a window or restarts the browser. Users will still be able to switch from the custom toolbar, the IE standard toolbar, the Netscape-compatible toolbar, and the "compatible-plus" toolbar. If the user rearranges the buttons while using one of the preconfigured toolbars, those changes will be saved to the preconfigured toolbar that is currently in use.

Customizing toolbar buttons

On the Mac, since we will have a finder-type view of the buttons in the toolbar button palette, users will be able to change button names, just like in the Finder. As a "power" feature, users will also be able to paste in different icons. (Many mac users know that you can change the icon of a folder, etc., by clicking Command I, then cutting and pasting the icons)

---

user assistance and discoverability

We are still debating over where to put this on the menus, but the user should be able to discover this feature in the following ways:

- menu item
- contextual menu item
- Command key (probably command plus clicking and holding on the toolbar buttons)

---

compatibility and platform requirements

Office Mac already has code that allows for customizable toolbars. Mac IE will not be able to leverage the code, but we should take the overall concept and simplify the UI; IE Win32 may choose to actually leverage the Office code, but there needs to be some synch up between Mac and Win IE in terms of how this looks to users.

---

setup and administration

For IEAK, etc., we will need to worry about having administrators customize the toolbar and locking down settings (i.e., eliminating the Edit toolbar command for users). Also, there may be branding issues in terms of customizing icons.

---

q&a

none.

---

open issues

Issue: Need to continue to explore toolbar persistence, how many toolbars people can have, how changes get saved when you rearrange buttons, etc.
Issue: Need to rationalize with OE (i.e., try to make it so they can leverage our code) and with Office.

II. Download Bar

summary

This feature allows users to view and manage multiple ftp downloads using an Explorer bar. Users drag a link to the download bar to download, and live status information is displayed on the bar.

design goals and justification

Simplicity: A popular daily-to-weekly browsing task is downloading files (pictures, software, etc.) to disk. Currently, IE for Mac has a download manager that has been very well-received by customers, because it allows users to monitor all of their ftp downloads in one window, showing current status while allowing users to go on with their other browsing activities at the same time. To make this even easier for users to access, we have proposed turning the download manager into an explorer bar, so that users will no longer have to switch to another window to get detailed status on their downloads.

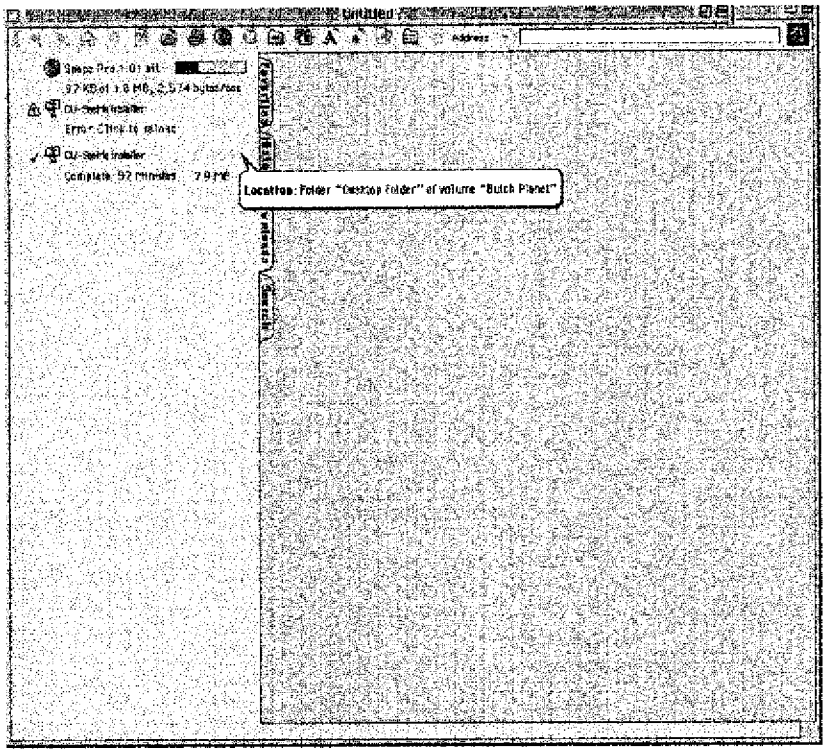
scenario

A possible scenario:

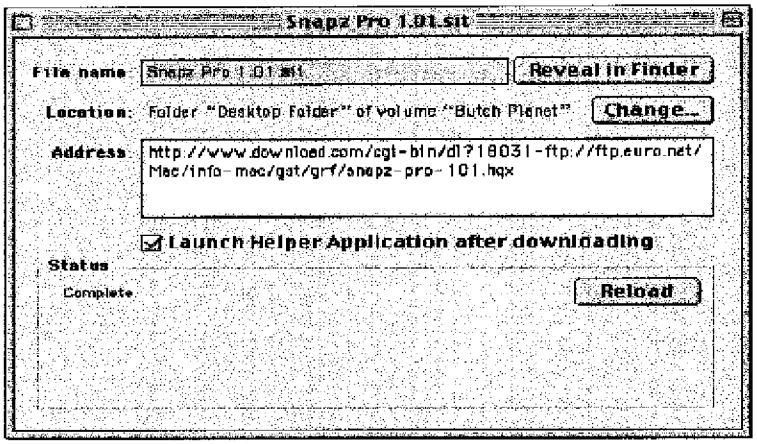
- John Smith administers a small Mac workgroup at his company. On a daily basis, he scours the web for browser and OS software updates; also utilities that may make his life easier. He likes to set a bunch of downloads in the morning and then go on with his daily activities. However, sometimes he comes back to his computer and finds that a couple of downloads had errors, so he has to go all the way back to the original web page and reload. Also, sometimes he has so many downloads that he forgets where he downloaded them to and must search his computer for the files (if the filenames are recognizable, which a lot of times they aren't).

design

When a user wants to download a file, they can simply drag the download link to the "Downloads" explorer bar. (When the user hovers over the download explorer bar tab, there will be a tooltip that says "Drag a link to download files"). The download bar (see below) will then open up to show file status.(The download bar will also open if the user downloads a link the "normal" way, by clicking on a download link.) Each file item on the explorer bar will contain filename, mini-progress bar, and status of bytes transferred. A successful download will be indicated with a checkmark, and an error with an exclamation point. Once the download is completed, users can hover over the item and view a tooltip that displays the location where the item was downloaded.



When users click on a download item in the download bar, the info window comes up.



This window shows much of the same information as in the download bar, but gives more detail (the url where the item was downloaded from, etc.) Users can reload from this window, and can also reveal the file in the finder if it was loaded to the desktop, change the location of the download folder when reloading, etc. Users will also be able to access a contextual menu when control-clicking on the item. This menu will allow users to reload, get info, rename, delete, etc.
Users can set in the Preferences how many downloads the download bar can hold, as well as a default download location. Users can delete a download item by dragging it to the trash.

Status alerts

When the user closes the download bar and goes about other activities, they will be able to set whether or not they want a sound or dialog alert to tell them when all downloads are complete. Also, the download bar tab will display an exclamation point, checkmark, or "in progress indicator" to indicate the status of one or more downloads in the download bar.

---

user assistance and discoverability

Since we are proposing to move the download functionality to the explorer bar, we hope that this feature will now be even more discoverable.

---

compatibility and platform requirements

We have shown the original download manager to the Win32 IE team, and are urging them to design a similar feature.

---

setup and administration

As monitoring the files in the download manager does not affect restriction of downloads via security zones or other options, we don't feel there are going to be any major issues, although the IEAK for Mac lead will be reviewing this spec.

---

q&a

none at this time

---

open issues

None at this time.

III. Mail Notification Within IE

summary

This feature allows users to receive notification within their web browser as to when they have received new Outlook Express messages. This mail notification feature will also allow users to open the first five of their new messages from within the browser. [would make a good explorer bar feature, or we could possibly just have users access from the mail menu, as shown in this spec.]

design goals and justification

Simplicity: Studies have shown (cf. some of the Win32 team's most recent IE usability studies) that the first three tasks that people accomplish when coming into work in the morning is check voice mail, check email, check favorite sites. By combining mail notification with web browsing, we are hoping to make accomplishing these tasks even easier.

scenario

A possible scenario:

- Sally Johnson spends most of her time at work using a web browser for research. She likes the fact that the email and web browsing are integrated in her browser; every time she wants to send a message, all she has to do is go to the "new message" item on her browser mail menu. She also likes that she can open her inbox from IE to read her new messages, but wishes that she could quickly scan her new messages from within the browser so she doesn't have to disturb her work by switching to her inbox every time she gets new mail.

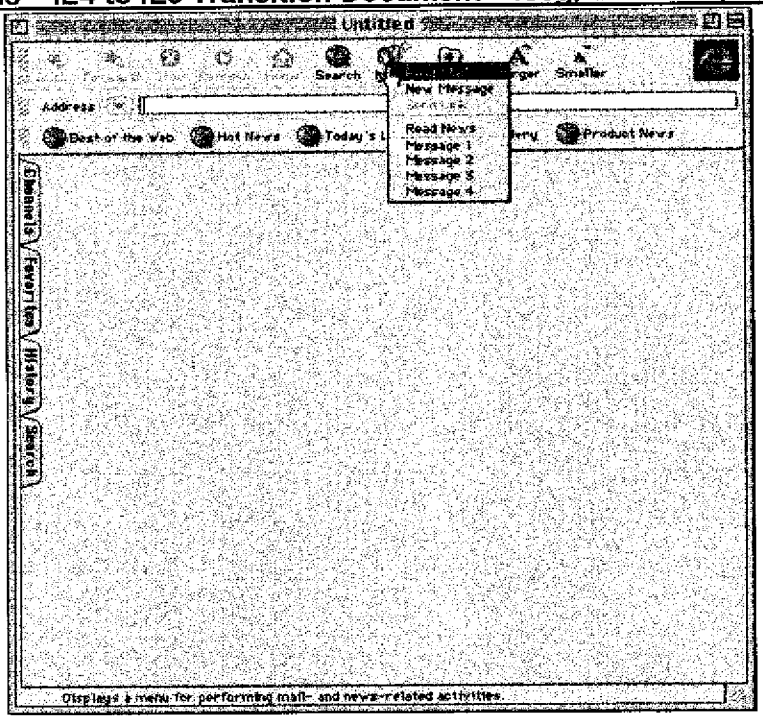
design

When the user gets new Outlook Express mail, the browser indicates this via a sound, or via a change in the Mail icon on the toolbar (users will be able to set their preference.) Since the Internet Services library where this information is stored is shared by both OE and IE for Mac, OE does not have to be running for this notification to occur.

IE can also grab the first few new message headers from ISL (probably around five would be good) and display them in the mail menu as shown above (indicated by "message 1," etc.) Thus, users will be able to quickly scan their messages in this menu; if they would like to open a message, they simply select it in the menu and OE will launch and open this message for the user. Another possibility is to have the message text actually show up in the browser itself. Either way, ISL could register whether or not the message is read or unread and pass that information to OE.

If the messages were displayed as links in an Explorer bar, the user might also be able to use a contextual menu to mark as read/mark as unread, etc.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc



---

user assistance and discoverability

Still under discussion as to whether this should be accessed from the toolbar or from the Explorer bar.

---

compatibility and platform requirements

Currently, the Win32 team is not planning on implementing a mail notification feature. We need to discuss whether this is something we want to add to the Mac product.

---

setup and administration

No issues at this time.

---

q&a

none.

---

open issues

Issue: Explorer bar vs. Mail button menu.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Lee's Transition Information

My areas of Mac IE5 ownership are as follows:

- ***TCO*** - Overall ownership for efforts to reduce the Total Cost of Ownership - this spans the entire product.
- ***IEAK*** - The Internet Explorer Administration Kit.
- ***ICW*** - The Internet Connection Wizard.
- ***Active Setup*** - The internet based installer technology we introduced with Mac

It is worth noting that the above areas are generally not supposed to match Win32 feature for feature. But rather only those features which are justified to the Mac Platform.

For IE 4 I have owned:

- ***Release*** - The actual RTW process of the product at launch time.

TCO:

For the past 2 months I've been attending bi-monthly meetings in Redmond with Timothy Johnson of the Redmond TCO team. These meetings have at times included the Unix TCO PM along with various members of the Win32 TCO team. They have been working on their planning for IE5 and are coding on their features at this time.

TCO for Mac IE has not yet been scoped. It has been unclear as to which TCO features would appropriately be brought into the platform. However, a dynamic PID has been a strong request from PSS and should be considered required at this time.

Additional features to be decided after visiting top 10 Mac Corps/Edu sites, and revisiting Win32 features.

IEAK:

IEAK 5 planning for the Mac has consisted of adding support for AutoProxy both to the IEAK as well as the app itself. The specs for this are from the win32 team and are available at http://ie. This is a strong ask from the field for corps/edu support and is a well understood technology. The following areas are to be considered for feature work:

- Support for new Open Transport (integrated control panel and preferences).
- Support for the Location Manager.
- New AppleGuide UI.
- Integrate default config. ICP and help files into ICW resource fork (All in one ICW app).
- Interface to enter MacPPP and OT/PPP script or script file information
- Icon Animator integrated into the wizard

Additional features to be decided after visiting top 10 Mac Corps/Edu sites, and revisiting Win32 features.

ICW:

ICW 5 planning has consisted mainly of modifying the UI to be consistent with the Apple Assistant look and feel. We are already nearly done with this work, as we are delivering this to Apple for their Allegro release. The only open item remaining would be doing the work to move to the new MS Referral Server (completed with the launch of win98). We expect to receive feedback from the Allegro team during their development cycle and more feedback from customers post Allegro ship.

Additional features to be decided after visiting top 10 Mac Corps/Edu sites, and revisiting Win32 features.

Loose Ends – IE4 to IE5 Transition Document – leeg, debweiss, hillelc

Active Setup:

AS 5 planning is largely bug fixes along with possibly adding support for changing channel content on the fly. That is to be able to download different channel information based on a file on the ms.com download servers. A larger issue of moving the AS technology to different portions of the product is in need of revisiting - use it for IEAK as Win32 does, use it exclusively for installs - no preconfigured installers?

Additional features to be decided after visiting top 10 Mac Corps/Edu sites, and revisiting Win32 features.

Release:

Releasing the IE/OE products consists of the following:

- Filling out release pages on http://rtw/release/tool.htm. This is the means by which all web releases are done. This is the same group that handles traditional media releases - WPRS.
- Getting content information to the IE Web team so that they are able to produce both content pages along with download instructions.
- Coordinating with local QA as to when they have a final candidate ready. Working with the ms.com team to propagate that build early so that the QA team can test. This is important since Active Setup relies on files on ms.com to be there in order to work right. Thus it is a very different release for the ms.com team.
- Once QA signs off on the build, get the ms.com team to prop the binaries if they haven't already. Then download all files from all sites. This is a long and tedious process, but we have been burned before when files have not shown up on sites, but the download pages are live.
- Prop binaries early if at all possible! This will mean that only download pages must be propped the night before release, and a much lower risk of failure. This is possible since all download servers are http:, and thus people won't find the files so long as you place them into a new directory or give them different names.