From: James Plamondon
Sent: Tuesday, January 11, 2000 4:32 PM
To: Peter Plamondon
Subject: FW: Windows Evangelism!

Effective   Evangelism is War.ppt   Generalized   Power Evangelism.ppt   tips.htm
Evangelism.doc   Evangelism Timelin...

——Original Message——
From: Peter Plamondon
Sent: Tuesday, January 11, 2000 2:58 PM
To: James Plamondon
Subject: RE: Windows Evangelism!

Please forward those docs to me as well. --Peter

——Original Message——
From: James Plamondon
Sent: Tuesday, January 11, 2000 2:47 PM
To: Peter Plamondon
Subject: FW: Windows Evangelism!

"DRG is asleep at the wheel"

——Original Message——
From: Darryn Dieken
Sent: Thursday, January 06, 2000 12:25 AM
To: James Plamondon
Subject: RE: Windows Evangelism!

Thanks James. I'll be thinking of you sitting on the beach bathing in the sunlight when it's dark and rainy here in Seattle.
Drink a pina colada for me :-)

ps. I think it's fair to say that DRG is asleep at the wheel. They only have one person evangelising Windows technologies.
Do you know of anyone else around the company who'd be interested in joining John new team?

——Original Message——
From: James Plamondon
Sent: Wednesday, January 05, 2000 5:33 PM
To: Darryn Dieken
Cc: John Colleran
Subject: RE: Windows Evangelism!

Darryn —

Sounds great! But I'm leaving the company in early March, and indeed will be on vacation until then, starting at the
end of next week. Upon departure from Microsoft, I'm moving my family to Australia, where I will sit on the beach,
drink pina coladas, and laugh at the world.

Here are some documents from the Good Old Days that you might find to be handy in starting this new group. Don't
worry about them leaking; they were already entered into the public record by Bristol Technologies as part of their
private anti-trust case.

<< File: Effective Evangelism.doc >> << File: Evangelism is War.ppt >> << File: Generalized Evangelism
Timeline.doc >> << File: Power Evangelism.ppt >> << File: tips.htm >>

Hoping that these will help you keep the flame alive after the old war-horses like me go out to pasture, I remain

Yours,

EXH. 6 DATE 5/24/0?
WITNESS Plamondon
SUSAN ZIELIE

MS-PCA 1913141
HIGHLY CONFIDENTIAL

James

P.S.: There are only FIVE certified Windows 2000 apps? Five? No wonder you're starting your own evangelism group! Is DRG asleep at the switch, or what?

<http://www.zdnet.com/zdnn/stories/news/0,4586,2415432,00.html>

...and if this article is incorrect, then shame on DRG for having done such a poor job of getting the accurate (and presumably higher) numbers out there!

-----Original Message-----
From: Darryn Dieken
Sent: Wednesday, January 05, 2000 11:51 AM
To: James Plamondon
Cc: John Colleran
Subject: Windows Evangelism!

Hey James,

Long time no see -- hope you're doing well and enjoying you're sabatical in MSR :-) JohnC is starting a new Windows evangelism team under BrianV and I thought you may be interested. The group is still being formed so there's a lot of opportunity for an uber-evangelist like yourself!!! I thought you may be interested -- let John know if want learn more about his new team.

--Darryn

Effective Evangelism

James Plamondon, Technical Evangelist

Evangelism Is War

Our mission is to establish Microsoft's platforms as the *de facto* standards throughout the computer industry. Our enemies are the vendors of platforms that compete with ours: Netscape, Sun, IBM, Oracle, Lotus, etc. The field of battle is the software industry. Success is measured in shipping applications. Every line of code that is written to our standards is a small victory; every line of code that is written to any other standard, is a small defeat. Total victory, for DRG, is the universal adoption of our standards by developers, as this is an important step towards total victory for Microsoft itself: "A computer on every desk and in every home, running Microsoft software."

Our weapons are psychological, economic, and political—not military. No one is forced to adopt our standards at the barrel of a gun. We can only convince, not compel. Those who adopt our standards do so as a rational decision to serve their own ends, whatever those may be. It is our job to ensure that those choosing an operating system are presented with an overwhelming abundance of evidence and reasoned argument in favor of our standards—so overwhelming that the choice of our standards seems obvious, or (ideally) that the developer is not even aware that a decision was faced, and a choice made.

We do this by understanding the barriers that might otherwise prevent the developer from adopting our standards, and removing them; by understanding the inducements that might facilitate the developer's adoption of our standards, and providing them; by understanding the arguments of our competition, and countering them.

Our Mission

The Charter of Microsoft's Developer Relations Group is clear:

> *Drive the success of Microsoft's platforms by creating a critical mass of third-party software applications and business solutions.*

This mission statement contains both the goal which we are striving to achieve (the success of Microsoft's platforms) as well as the means by which we are to achieve it (by creating a critical mass of third-party applications and business solutions). Let us examine it carefully.

Platforms

What is a platform? A platform is any software service which an ISV's code assumes is present in order for it to function properly. For example, Windows is a platform, because ISVs can write code that assumes the presence of the Windows DLLs, and which will not function properly in their absence.

What are Microsoft's platforms? Once, the answer was simple: Windows (and before that, DOS). Now, Microsoft has many platforms, and the answer is more complicated. Windows is still one of Microsoft's platforms. New platforms include Microsoft Office, Microsoft Back Office, Microsoft Exchange, The Microsoft Network, Microsoft Internet Explorer, DevOffice, and ActiveX. For each of these, ISVs can (or will be able to) write code that assumes the platform's presence.

Turning a product into a platform perpetuates and broadens the success of the product. This is because writing code to support even one platform is hard; writing code to support multiple, alternative platforms is

exponentially more difficult. This leads developers to choose between alternative platforms, and (ideally) choose one over all others. This choice, once made, is not easily changed; code written to one platform is only re-written to another with difficulty. In effect, once a developer chooses a platform, the developer is "locked in" to that platform.

Critical Mass

"Critical mass" is a term borrowed from nuclear physics, in which it means "a sufficient mass of fissionable material to sustain a chain reaction," in which a chain reaction is defined to be "a self-sustaining nuclear reaction yielding energy that causes further reactions of the same kind."

"Self-sustaining" is the key phrase in this definition. Once started, the nuclear reaction naturally re-invests its energy in perpetuating and growing the reaction. Once the point of criticality is reached, no additional external energy or intervention is required to maintain the reaction.

So it is with evangelism. Once a platform has reached a certain degree of support in the industry, it becomes self-perpetuating (for reasons discussed below). Once this "critical mass" of support for a given platform has been achieved, new applications will tend to support it too, even in the absence of further evangelism.

Note that were are concerned with the "mass" of support by. Both the number and the importance of applications supporting a platform contribute to the overall mass of the platform's support.

Mass in motion gives momentum, which can be defined as the amount of effort required over a given time to stop a body in motion. A sufficient mass of applications delivering support for a platform over a sufficiently short time, gives the platform sufficient momentum that it is, for all practical purposes, unstoppable.[1]

Current economic theory[2] suggests that the emergence of a dominant standard from a number of competing alternatives, is the result of "random events." It is our job to ensure that there's nothing random about it: we create the platform momentum that ensures Microsoft's victory.

Third-Party Applications and Business Solutions

Platforms, by themselves, are of little or no value to end users. Their value is derived from the applications and business solutions which take advantage of the platforms' services in order to deliver real value to the customer. If we wish our platforms to be successful, we must ensure that our platforms are used by the applications and business solutions that customers want.

Third-party applications[3] and business solutions are the measurement of the success of evangelism. The greater the mass of shipping applications that supports our platforms, the more successful we have been.

The art of evangelism is concerned with increasing the mass of applications and business solutions, that support our platforms, to the point of criticality.

---

[1] It is arguable that it is this unstoppable momentum, not just sufficient mass, that is our objective; however, since unstoppable momentum is an inevitable consequence of attaining critical mass, the distinction is moot.
[2] "Increasing Returns" theory, promulgated by Brian Arthur of Stanford.
[3] Historically, DRG has not evangelized the use of our platforms by Microsoft's own applications and business solutions. It is assumed that they are fully supporting our platforms. But as our technologies have become more numerous and complex, the need for internal evangelism has increased. Such internal evangelism can be highly leveraged, because our applications play an important role in setting standards in the industry. When the internal evangelism will clearly further DRG's ends, do it; but remember that at present, internal evangelism is not itself in DRG's charter — so it's a means to an end, not an end in itself.

Our Enemies

Some have claimed that Microsoft has a monopoly on the market for operating systems designed for personal computers. This is patently untrue. IBM's OS/2, Apple's MacOS, Novell's Unix and NetWare, Taligent's CommonPoint, Netscape's Navigator, and other operating systems (or embryonic operating systems), all compete with Windows in the struggle for applications and business solutions. This is true for our other platforms as well (Exchange/Notes, OLE/OpenDoc, Office/SmartSuite, etc.).

Nor is this a static list. The computer industry is very dynamic; we in evangelism must be even more dynamic – constantly searching for new allies, enemies, and strategies.

It is beyond the scope of this paper to expound on the strengths and weaknesses of each of our competitors. There are many other sources of such information. Use them!

The Field of Battle

The software industry is the field on which evangelists do battle. The industry has its own trade press, book publishers, developer SIGs, conferences, tradeshows, training organizations, retailers, wholesalers, and distributors. It has its own culture. Understanding the terrain of the industry is essential to effective evangelism.

It is not effective to publish an article about Mac OLE in a Unix magazine, or to give a talk on Win32's memory-mapped file I/O at a conference attended by executives. To be effective, the right tools and messages must be delivered to the right place, at the right time.

Some ISVs and their applications are more important to the success of a platform than others. They are the high ground of the battlefield. Gaining the support of a major ISV is like taking a hill; from there, you command the battlefield below. The same can be said for the trade magazines, conferences, developer SIGs, etc.. You cannot control the field of battle if the high ground is in the hands of the enemy.

The Measure of Success

The success of evangelism is measured in the mass of shipping applications that support Microsoft's platforms. Those applications that support a given platform, move it closer to critical mass, and thus count as successes; those that support a competing platform, move it away from critical mass, and thus count as defeats.

Microsoft has many platforms, and what appears to be a victory for one Microsoft platform may appear to be a defeat for another. If OLE is successful on the Macintosh (making Macintosh applications more powerful), is that a victory for OLE, or a defeat for Windows? If the Windows API is successful on Unix (making Unix applications less expensive to develop), is that a victory for the Win32 API, or a defeat for Windows NT?

Diversionary attacks, holding actions, and retreats may each seem contrary to the achievement of the overall objective when considered solely in their own terms, but taken in light of the overall conflict, may contribute to overall success. In the Chinese Civil War that followed World War II, Mao Tse Tung's Army ran away from every battle, until they won the war. They knew that overall victory, not local victory, was the objective.

Thus it is imperative to measure each action in accordance with its contribution to overall, not just local, victory.

Victory

"A computer on every desk and in every home, running Microsoft software." This is the mission statement of Microsoft itself; it is the definition of the conditions under which Microsoft itself can declare overall victory.

Definition of Evangelism at Microsoft

"Evangelism is the art and science of getting developers to ship products that support Microsoft's platforms."
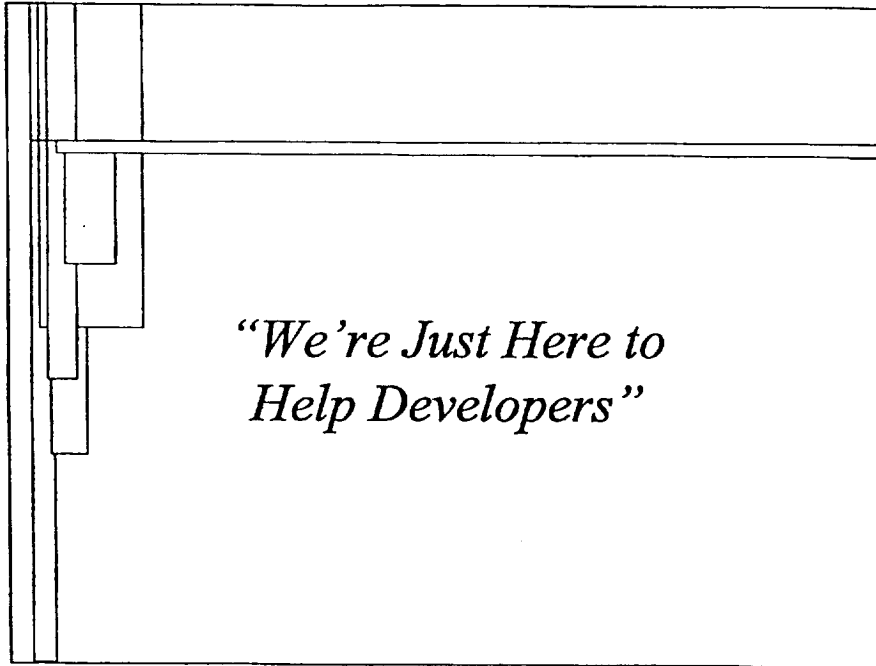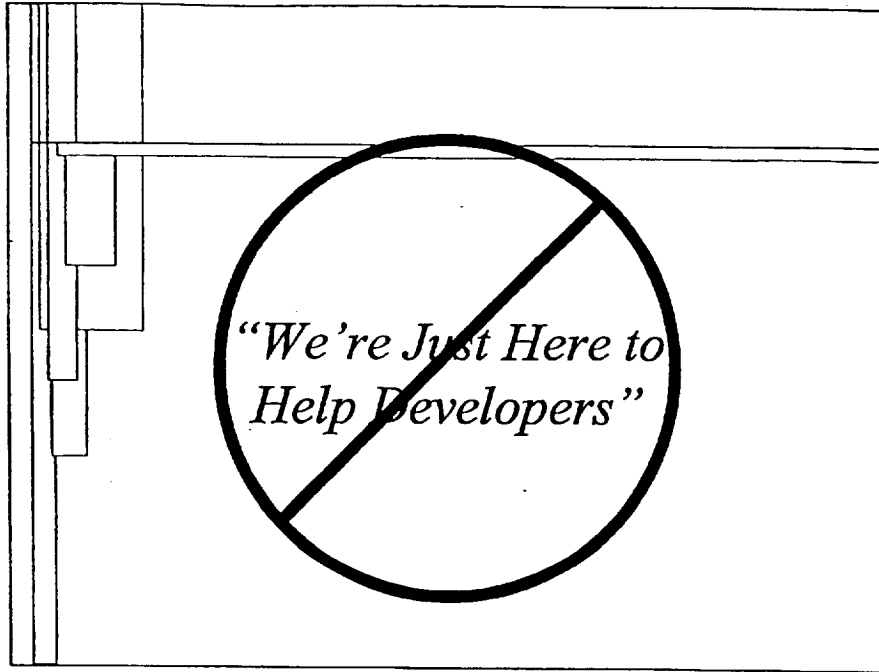
Effective Evangelism

James Plamondon
Technical Evangelist
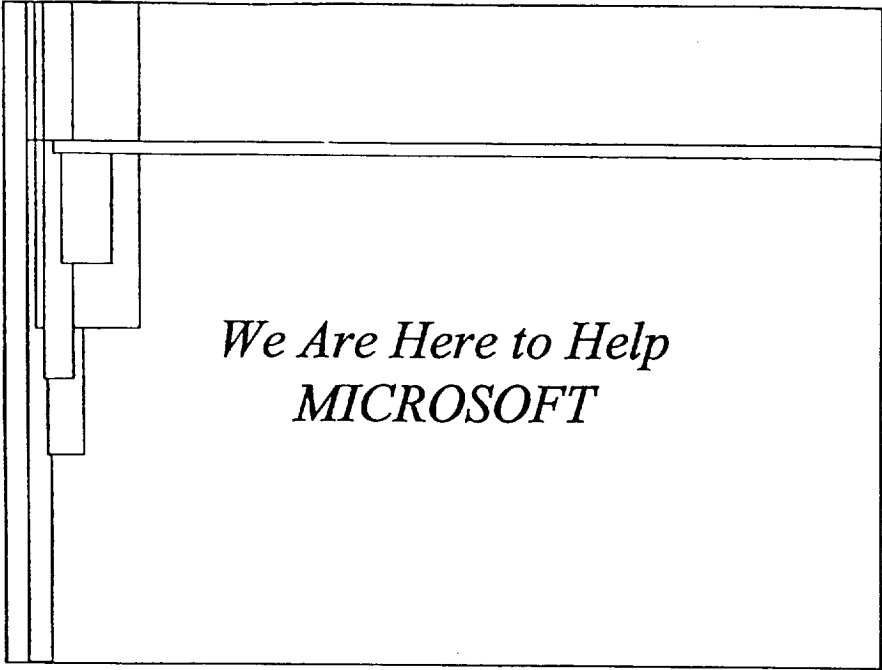### Microsoft Developer Relations Group

Mission Statement

*"Drive the success of Microsoft's platforms
by creating a critical mass
of third-party software applications
and business solutions."*

"We're Just Here to Help Developers"

3

3

We Are Here to Help
MICROSOFT

Evangelism is *WAR!*

- **Mission**
  - Establish Microsoft's platforms as de facto standards
- **Enemies**
  - Other platform vendors
- **Battlefield**
  - ISV Mindshare
- **Progress**
  - Shipping ISV applications

Role of ISVs

- *ISVs are just pawns in the struggle*
  - *But have you ever tried to win a chess game without any pawns?*
- *We need the support of ISVs to win*
- *We must <u>earn</u> the support of ISVs, by*
  - *Shipping technologies worthy of support*
  - *Helping ISVs implement and market their support for our technologies*
- *We work hard to help our ISVs succeed!*

So, We're Just Here to Help Developers, Right?

We're Here to Help Microsoft!

- **Microsoft** *pays our wages*
- **Microsoft** *provides our stock options*
- **Microsoft** *pays our expenses*
- *We're Here to Help* **Microsoft**
  - *By helping those developers...*
  - *...That can best help* **Microsoft** *...*
  - *...Achieve* **Microsoft** *'s objectives.*
- *Did anyone miss the point, here?*

So, How Do We Accomplish This Mission?

Mission Statement

"Drive the success of Microsoft's platforms
by creating a critical mass
of third-party software applications
and business solutions."

Mission Statement

"Drive the success of Microsoft's <u>platforms</u>
by creating a <u>critical mass</u>
of <u>third-party software</u> applications
and business solutions."

Mission Statement

"Drive the success of Microsoft's <u>platforms</u>
by creating a critical mass
of third-party software applications
and business solutions."

What is a "Platform"?

- Software service required by an ISV's app
- Windows, OS/2, Mac OS
- OLE, OpenDoc, CORBA, [D]SOM
- Exchange, Notes
- Office, SmartSuite, PerfectOffice
- BackOffice, Oracle, AS/400 apps
- Java, Navigator, the Web...ActiveX

Platform Example: Win32

- **Win32 on Windows**
  - *Windows NT, Win32s, Windows 95*
- **Win32 API on Unix**
  - *Sun, WABI*
  - *Bristol, MainSoft*
- **Win32 API on Mac**
  - *Visual C++ for Macintosh*
- **Result: Win32 is the Universal API**
  - *We Win!*

Mission Statement

"Drive the success of Microsoft's platforms
by creating a _critical mass_
of third-party software applications
and business solutions."

Critical Mass

- ■ "A sufficient mass of fissionable material to sustain a chain reaction"
- ■ Chain reaction
  - • "A self-sustaining nuclear reaction yielding energy that causes further reactions of the same kind."
- ■ Result
  - • Physics: Nuclear explosion
  - • Evangelism: De facto standard

Nukes 'R Us

- We create chain reactions
- Fissionable Material: Shipping Apps
  - Mindshare
  - Market Share
- Other Components
  - Timers, plastique, casing, delivery systems
  - Sample code, white papers, presentations, books, articles, meetings, press releases
- Blow the competition away

Self-Sustaining

- **Platform is**
  - *Expected by reviewers*
  - *Supported by tools, books, consultants, trainers*
- **".. yeilding energy that causes further reactions of the same kind."**
  - *It's evangelizing itself*
  - *So Microsoft can stop evangelising it*
- **Victory!**

Mission Statement

"Drive the success of Microsoft's platforms
by creating a critical mass
of _third-party software_ applications
and business solutions."

Third-Parties

- **Not Microsoft or its competitors**
  - First parties
- **Not our customers**
  - Second parties
- **Other software vendors**
  - Third parties

Why Not Do It All Ourselves?

- **Because we can't**
  - *There's just too much to be done*
- **Because they won't let us**
  - *Lawyers 'R Us*
- **Because third-parties are more efficient**
  - *In their respective markets*

Enlightened Self-Interest

■ We help ISVs to help ourselves

■ But we really do help them

■ We fight for our ISV's

- APIs, hooks, tool support
- Design Reviews
- Strategies, timelines, etc.
- ...because we need their support for our own ends

Too Many to Help

- Can't help 'em all
- We help those who can help _us_.
- If they can't or won't help us
  - Screw 'em!
  - Help their competitors instead

Summary

- We're here to help Microsoft...
- ...By helping those ISVs...
- ...That can help Microsoft...
- ...Achieve its objectives.

Strategy

Or, How to Kick Ass Without Breaking a Sweat

Strategy

- *"To win one hundred battles is not perfection; to subdue the enemy without fighting is perfection."*

Sun-Tzu: The Art Of War, written in China by Sun Wu in roughly 490 B.C.

Strategy of Sun Tzu

- *"Thus, the best military strategy is to attack the enemy's plans.*
- *Next best is to disrupt his alliances.*
- *The next best is to crush his army.*
- *The worst policy is to attack his fortified cities. Attack cities only when there is no alternative."*

Attack the Enemy's Plans

- Do not attack directly
  - No debates, no white papers, no lawsuits
- Do the unexpected; attack his assumptions

Example: Win32 API on Unix

- **Sun's plan**
  - *Implement Win32 on Unix (WABI)*
  - *Unix would run Windows software*
- **Assumption: MS would try to block WABI through legal action**
- **Microsoft's response**
  - *Put Win32 on Unix ourselves*
    - *Through Bristol, Mainsoft, Locus, etc.*
    - *They pay us a fee and a royalty*
  - *Don't sue; compete!*

Crush The Enemy's Army

- **Head-to-head evangelism**
  - *White paper wars*
  - *Panel debates*
  - *Technology duels*
- **Both sides lose**
  - *Credibility*
  - *Opportunity cost*
- **Avoid this strategy!**

Example: OLE vs. OpenDoc
The Old Strategy

- Direct competition on tech details
- Press, pundits LOVE this!
  - Conflict!
  - Underdog!
- ISVs are confused, so they do nothing
  - "I'll wait to see who wins"
  - OpenDoc's FUD worked
- Complete disaster
  - Delayed the widespread adoption of OLE

Disrupt The Enemy's Alliances

- **Alliances are weak**
- **Circumstances change over time**
  - *Technology, personnel, regulation, mergers*
- **Once-mutual interests, diverge**
  - *AppWare Foundation*
- **Each ally schemes for advantage**
  - *And to minimize its costs*

Example: OLE vs. OpenDoc
The New Strategy

- Allies
  - Apple, Novell, IBM
- Strategy
  - Disrupt the alliance
- Tactics
  - Reposition OpenDoc as an OLE dev. tool
  - Put OLE in the Enterprise
    - pits Apple against IBM & Novell
  - Help Claris & WP support OLE in Win95
    - pits part of an ally against itself

34

Attack The Enemy's Fortified Cities

- **Big ISVs that compete with Microsoft**
  - *Lotus, Novell, Oracle, ...*
- **They hate us**
  - *and there's nothing we can do about it.*
- **Don't throw yourself against their walls**
- **Help their competitors instead**
  - *Let them attack the cities for us*
  - *They'll be grateful for our help (for a little while...)*

Victory Conditions

- **You win when the enemy quits**
  - and not before
- **What will cause the enemy to quit?**
  - Lack of support
  - Public humiliation
  - Low return on investment
  - Nothing
- **Don't start fighting, until you can identify when you've won**

All Is Not Fair

- We are under close scrutiny
  - Any unethical acts WILL BE uncovered
- Besides -- we're the good guys!
- Simple rule to live by: _Never Lie_
  - Tell the truth, and nothing but the truth
  - Be selective in which truths you emphasize
  - Let the competition fill in the gaps

Next Time

■ *Power Evangelism*

- *Definition of Power in Evangelism.*
  Resource exchange. Gaining power.
  Using power. Getting things done.
  Leverage. Influence. Mind Control. Tactics.

Generalized Evangelism Timeline

Not To Scale

```
                S1   A1              B1          9                      10              V
----------------------------------------------------------------------------------------------
1    22222      3    4    5    66666666666   7   888888888888888888888888888    11-11-11-11   12
```

1: *Evangelism starts.* Goals, objective, strategy & tactics, ISVs, and influencers, are defined. Verify that spec & demos are nearing stability. Verify PSS & dev team commitment to support ISVs in DevLab. Identify target ISVs, and the potential carrots & sticks that you can use to motivate them. Think about development tool issues (most other ISVs will need tools to implement support for your technology; what tool products are critical? Target them first).

2: *DR Prep.* Preparations for a Design Review (DR) begin. Line up speakers, logistics, invitations, skeleton press release, "showcase" ISVs, etc.

3: *Host DR.* Upon availability of (a) a spec (S1) and/or (b) first internal build stable enough to demo, host the targeted ISVs at a Design Review. Goals: (i) verify that the technology is useful to ISVs; (ii) get ISVs emotionally involved in the technology; (iii) generate positive press regarding the technology.

4: *Limited Developers' Release.* Technology & specification reach sufficient stability (A1) to be released to the targeted developers. (Stages 3 & 4 are often collapsed together, such that the first stable release is given to the Design Review attendees.) This is *not* a general beta release.

5: *Jihad.* At (or soon after) the Design Review, send ISVs the details of the [Technology Name] FirstWave Program, including the Letter of Agreement, for their review. Within a few days, visit each targeted ISV in person, to close their participation in the FirstWave Program.

6: *FirstWave Program.* Host the targeted ISVs in a series of visits (rarely more than three) in the DevLab, focused on producing versions of their apps that demonstrate the ISV's support for [Technology Name]. Monitor progress of ISVs towards successful demonstration; send details up the management chain. Consolidate ISV feedback, bug fix requests, etc., and generally represent the ISVs to the technology product team. Ensure that adequate sample code and documentation is being produced. Line up courseware, books (MS Press and third party), articles, conference sessions, stacked panel discussions, seminars, magazine articles, defections from the competition, rifts in enemy alliances, etc.

7: *Beta Release.* Upon the wide release of a stable beta of the technology to large numbers of developers (e.g., via MSDN or the Web), host an event for ISVs, press, analysts, allies and competitors, at which the targeted ISVs demonstrate their apps' support of the technology, as produced in the DevLab.

8: *The Slog.* Work with early adopters to help get their products ready to ship. Represent the ISVs at bug-prioritization meetings. Speak with press, analysts, allies, and competitors. Broaden evangelism efforts through the involvement of MSDN's manifold resources. Execute the tactics as previously determined, with minor modifications on the fly; this is the battle phase, win or lose.

9: *Final Release.* Technology is released in final form. Big event for press, analysts, ISVs, allies, competitors, etc., showing momentum – early adopters demo apps (which are hopefully at or near final themselves), plus demos of other cool apps that turned up during the Slog. The Slog may continue for some time.

10: *Critical Mass.* The technology is recognized as the *de facto* standard, and the vast majority of developers incorporate it into their future plans. Some influencers may actually increase their support for competing technologies, but it's too late – critical mass has been achieved, and they are at Ground Zero.

11: *Mopping Up.* During the mopping-up phase, ensure that the enemy technology is routed. Use the press, the Internet, etc. to heighten the impression that the enemy is desperate, demoralized, defeated, deceased. Usually, this phase (or even Phase 8, the Slog) overlaps phases 1-3 of the next version of [Technology Name], which addresses all of the advantages of the competitors' technology, while addressing [Technology Name]'s key weaknesses. Repeat phases 1-10 as necessary (e.g., Windows NT 3.1 through 4.0; OLE 1.0 through COM+; Windows 1.0 through Windows 3.1).

12: *Victory.* The developers, marketers, and managers of the competing technology give up the sinking ship, and interview for positions at Microsoft.

Generalized Evangelism Timeline

James Plamondon
Technical Evangelist
Microsoft Developer Relations Group
October 9, 1997

Details

1: Evangelism Starts

Work closely with the technology development team, to ensure that they are developing the right technology, the right way. The right technology is that which will make money for Microsoft, or prevent Microsoft from losing money, whether directly or indirectly. The right way to develop technology is with a clear vision of how the technology is going to meet that objective – with a clear, concise focus on what actions need to be taken – by whom, when, how, and for what reasons – in order to achieve that objective.

One factor – but by no means the only factor – in verifying that the technology being developed is the right one, is ensuring that the technology meets the needs of any ISVs that we will need to have support it. If the technology can succeed without third-party support, then this is not a consideration, and Evangelism will be relatively uninvolved in the process. On the other hand, if ISV support is a key factor in the success of the technology, then Evangelism's role will be critical.

Well before there is a detailed spec, decide (with the technology team) which ISVs' support is (a) most critical to the success of the technology, and (b) most easily gained, for whatever reasons. Under NDA, and with the support of the technology team, discuss the technology informally with a small number (four, plus or minus two) of these critical ISVs, to get a "reality check" on ISV reactions to the technology. Make sure that the feedback from these ISVs is either incorporated into the spec, or discarded with good reason. If the feedback is discarded, formulate compelling Q&A explaining the reasons for discarding it. Run the results past your small number of ISVs, to verify that it is compelling. If not, rethink the discarded feedback. If the overall feedback is negative, rethink development of the technology.

Now, you are prepared to define the evangelism plan. First, define the goal of the technology. This is usually quite simple: to become the *de facto* industry standard mechanism for [accomplishing some task]. To achieve this overall goal, one must achieve a number of specific, measurable objectives along the way. Some examples of objectives might be
- Having n tool vendors commit to shipping support for [technology name] in their tools by [some specific date]
- Having m ISVs demonstrate support for [technology name] in beta versions of their apps at [some specific venue]
- Having y books on [technology name] be available by [date], to help developers implement support for [technology name] in their applications

Presuming that ISV support is important to the success of the technology, then there usually exists an essentially infinite number of ISVs that could support the technology. However, Evangelism usually lacks the resources necessary to work closely with all of these ISVs. Further, some of these ISVs will contribute much more to the success or failure of the technology than others. Therefore, Evangelism usually must select a subset of ISVs to work with, out of the complete set of ISVs with whom it could work.

Those ISVs should be selected for evangelism, which will deliver the most benefit for the least amount of work. Some will be selected because they are easy wins – a company that has expressly and joyfully tied its success very closely to our own, or a company which is run by an ex-Softie who misses working with Microsoft. Some will be selected because they are the most influential, either in terms of market share or

mind share. (Mind share denotes the extent to which a product is discussed; market share denotes the extent to which a product is used.)

The ISVs targeted for evangelism should be categorized into four tiers:
1. Tier A: The most influential ISVs, in terms of market share or mind share. These will usually be obvious, but always do some research anyway to make sure that you're not overlooking anyone.
2. Tier B: Less influential ISVs, who might be worth evangelizing anyway – perhaps because they are
   a) Easy wins
   b) Generally assumed to be in the enemy camp, such that their supporting your technology would be a shock (and, hopefully cause a rift in the enemy camp)
   c) Utterly worthless evangelically, but you can lead the enemy to waste a lot of resources evangelizing them, if the enemy knows that "Microsoft cares a lot" about them.
3. Tier C: ISVs utterly lacking in influence, but whom you might speak to about the technology, if you can reach them in a one-to-many fashion – perhaps while you're visiting Tier A accounts in their city.
4. Tier Z: ISVs that should never know that you exist, lest they send you email, call you, or otherwise waste your time.

For some technologies, Independent Content Providers (ICPs) will be more relevant than Independent Software Providers (ISVs). For some, you may have to involve hardware vendors, advertising agencies, or other providers of evangelical leverage. I use "ISV" in this paper to generally denote all of these kinds of evangelism targets; make whatever substitutions are necessary for your specific technology, service, or whatever.

In addition to identifying and categorizing the relevant ISVs, Evangelism should also identify and categorize other industry influencers during this first phase of evangelism. There are three categories of industry influencers:
1. Providers of evangelical infrastructure: The authors of technical books, courseware designers and instructors, authors of technical articles, conference organizers, software consultants and contract development houses – Evangelism needs to involve all of these folks at various stages of an evangelism campaign, to make sure that ISVs and individual developers have the information they need to (a) decide to support the technology, and to (b) implement the technology in their application.
2. The Press: Almost every evangelism campaign involves working with the press, either directly or through a PR agency. Our evangelism plan should identify the specific members of the press that you will target for technical evangelism (as distinct from the usual, non-technical PR treatment).
3. Analysts: Analysts are people who are paid to take a stand, while always trying to appear to be disinterested observers (since the appearance of independence maximizes the price they can charge for selling out). Treat them as you would treat nuclear weapons – as an important part of your arsenal, which you want to keep out of the hands of the enemy. Bribe Hire them to produce "studies" that "prove" that your technology is superior to the enemy's, and that it is gaining momentum faster.

Having identified the objectives sought, the ISVs whose support is necessary to achieve that support, and the specific carrots and sticks that you are going to use to gain their support, you are prepared to document your proposed plan in an Evangelism Plan. A sample Evangelism Plan can be found at the end of this document.[1]

After the completion of the Plan, you are ready to prepare for the presentation of the technology to a group of ISVs: the Design Review.

2: Design Review Preparation

A Design Review is a gathering of evangelism targets, usually on campus, to whom Microsoft presents an early version of the technology. The goals of a Design Review are four-fold:

---

[1] Thanks to Peter Plamondon.

1. To verify that the technology is genuinely useful to ISVs, without any glaring weaknesses that will prevent its being used in the intended manner – and to gather any suggestions that might improve it
2. To make the attendees feel honored, respected, and involved, such that they become emotionally attached to the technology, and are therefore more likely to support it in their products
3. To act as a forcing-function for the technology team, by requiring them to produce clear descriptions, demos, and specs.
4. To generate positive press regarding the technology

To ensure that these objectives are accomplished at the Design Review, Evangelism must prepare extensively before the DR takes place. A date must be selected, to which the technology team will commit. Meeting room space must be reserved (usually in Building 12, or at a nearby hotel's conference center) and other logistics (meals, handouts, etc) arranged. A draft press release must be approved, pending final quotes from the attendees (gathered at the end of the DR). Speakers must be arranged, their slides reviewed, their demos tested, etc. – and the evangelist owns the whole shebang. The evangelist doesn't have to *do* it all, and indeed should do as little of it as possible. But the evangelist has to ensure that it all gets done properly, and on time.

Before the DR, various steps can be taken behind the scenes to ensure that the result is positive. Many of these steps are right out of books such as "What They Don't Teach You At The Harvard Business School," or manuals on winning at office politics, or the like. For example, you should always know what your ISVs are going to say at the DR, before the DR, so that you don't get any rude surprises. It is best (although not always possible) for the evangelist to meet personally with each DR attendee before the DR – perhaps hand-delivering the spec to be discussed at the DR. At this pre-DR meeting, the evangelist can go over the spec at a fairly high-level, and get a feel for what the ISV is likely to say or do at the DR. If the ISV's comments are likely to be positive, then the ISV can be encouraged to speak up at the DR, and the speakers can be encouraged to call on that attendee. If the ISV's comments are likely to be negative, the evangelist can try to have them addressed before the DR. They could be addressed, for example, by revising the spec – or perhaps by lining up other ISVs to oppose the negative comments at the DR (so that Microsoft is seen to be bowing to the will of other ISVs, rather than ramming its spec down the negative ISV's throat).

Also, you may want to work closely with one or two "showcase" ISVs, to get them to demo support for the technology at the DR itself. This is risky. On the one hand, it shows that the technology really works, can be implemented relatively quickly, and already has momentum – perhaps instilling fear in the other attendees of being "left behind" if they don't catch up quickly with the "showcases." On the other hand, it will irritate the ISVs that were not chosen to be "showcases;" they will feel slighted.

Invitations to DR's should be personalized, and sent out via email three weeks in advance, plus or minus a week or so. (Microsoft Word has a great mail-merge to email feature.) Any less warning and ISVs will have prior commitments. Also, our expecting developers to drop everything to come to us on very short notice, reinforces the notion that Microsoft is arrogant and demanding, among exactly the group that we most want to have thinking of us as being helpful and responsive. It is very insulting to ask someone to come out on less than two weeks' notice.

Invitations should go out to a friendly subset of Tier A ISVs and to providers of evangelical infrastructure (see above). Press should rarely be invited. Analysts should only be invited if they are already bought and paid for. PSS should send the staff that they have wisely assigned to cover this emerging technology.

3: Host the Design Review

The goals of a Design Review are described above. Design Reviews are relatively informal by nature – no one wants or expects Robert's Rules of Order to be followed. However, the presenters should be properly prepared, the demos should not crash (too often), the food should be good (and in the right quantity), etc.

The Evangelist should host the Design Review. The responsibilities of the host include
- Greeting ISVs as they arrive, and encouraging other evangelists to greet their accounts, too

- Making sure that the guests' random problems (rental car trouble, flight change issues, ripped pant seats, sudden illnesses, etc.) are dealt with promptly and with discretion
- Introducing each speaker, and making sure that each speaker sticks to the schedule
- Assisting with Q&A
- Schmoozing like mad during breaks – gathering feedback from as many attendees as possible, and pointing attendees with tough questions or great suggestions at the appropriate speakers
- Making sure that the speakers stick around for the breaks, meals, and evening events (if any)
- Gather written, confirmed quotes from the ISVs for use in a post-DR press release (if any)
- Generally doing anything else necessary (within the bounds of reason and taste) to ensure that the attendees get the information they need, give us the feedback that we need, and have a great time.

Design Reviews should be chock-full of juicy technical information, with a smattering of market opportunity analysis and business tips. While many hard-core technoids would rather read a spec than a novel, this is not universally true, so Design Reviews should provide some non-technical fun, too. If the DR runs for two days, have a booze-n-schmooze event at some fun location on the middle evening.

Most of all, the attendees should be constantly treated with respect. Each attendee should leave the event feeling that Microsoft really *listened* to what that attendee had to say.

Usually, an ISV will see that its competitors are also attending the DR. This is a good thing. When an ISV sees its competitors in the room, the ISV gets nervous that maybe its competitors are *already ahead*. Try to schmooze with ISVs within eyesight of their competitors. Every laugh and handshake you share with an ISV's competitor will strengthen the ISV's resolve to regain the lead.

Let the technology team spend their time explaining how the technology works. Spend *your* time listening to the ISVs explain what they think of the technology, and why. Take good notes, write them up with conclusions and recommendations, and forward them to the technology team.

Be sure that each attendee receives the message clearly and unambiguously: We need, want, and appreciate their support for this new technology.

4: Limited Developer's Release

After the design review – sometimes, at the design review itself – the technology will be sufficiently stable to release to a small group of developers, under NDA. The design review attendees should be the first to get this release, although others may also get it. The goals of this release are

- To generate feedback to the technology team on bugs, feature requests, documentation issues, and architectural issues that need to be addressed before the beta can go out to a wider beta group
- To help bring PSS personnel up to speed on supporting the technology
- To give book authors, consultants, contract houses, sample code writers, etc., something to get started with, so that they will be prepared to produce the necessary infrastructure when the technology goes into wider, non-NDA beta.
- To help Microsoft understand the strengths and weaknesses of our technology vs. that of any competitors, such that we can prepare technical, marketing, and business-development responses
- To support the argument that Microsoft is "open," given that Microsoft's internal product groups are almost certainly getting drops of the technology by this time.

Starting with the Limited Developer's Release, Evangelism should be sitting in on all PM and bug-prioritization meetings, prepared to present data that supports the adjustment of bug priorities and feature implementation order, based on ISV feedback.

Similarly, starting in this phase, Evangelism needs to be constantly monitoring the status of books, courseware, sample code, seminars, conference presentations, stacked panel discussions, and other elements of the evangelical infrastructure, to ensure that they are progressing towards availability at the time of general, non-NDA beta.

The Limited Developer's Release needs to be made available on a secure web server to which only participating ISVs have access. The server needs to be updated with new drops of the technology whenever they become sufficiently stable. Other means of electronic communication – private newsgroups/forums, list servers, bug & feature submission mechanisms, etc. – should be set up for use at this time, with an eye towards their scaling up as the number of beta sites increases over time.

The Early Adopters (or FirstWave) program for the technology is finalized during this period (initial carrots and sticks having been proposed in the very first evangelism phase). A FirstWave program offers specific Tier A ISVs the opportunity to get specific technical and marketing benefits in return for their commitment to delivering support for the technology in applications by a time. All elements of the program are laid out very specifically in a written document – a Memorandum of Understanding, also known as a Letter of Agreement or LOA. The LOA must be signed by an Officer of the ISV – a Director, VP, CEO, or President – to ensure that the commitment is taken seriously by the ISV. If a PM signs it, the ISV can always claim later that the PM didn't have the authority to do so. Similarly, the LOA needs to be signed by Microsoft's Director of Evangelism. LOA's are not legal, binding contracts, and thus give each party wiggle room. They need to be positioned as clarifying documents, that ensure that there are no misunderstandings as to what was being agreed upon, and by whom. We can't enforce an LOA, but we can certainly remember who has honored them, and who has not.

An alternative to an LOA is a legally binding contract. For most ISVs – with whom Microsoft has a long-standing relationship – a legally binding contract is overkill. For some ISVs – with whom our relationship may have been equally long, but somewhat more tempestuous – a legal contract may be required. There are other reasons why you might want to have a legal contract, but there's one big reason why you don't want to do so: because then you're going to have to deal not only with Microsoft Legal, but with every other company's legal departments, too. This will not be fun, and it will consume vast quantities of time that could be better spent on evangelism rather than legal wrangling.

A sample LOA can be found at the end of this document.[2]

It is imperative that Evangelism work closely with the marketing side of DRG in hammering out what marketing incentives can and cannot be promised to ISVs in the LOA. It's their budget, their marketing resources, and their time that we're promising. The Marketing folks have a lot of experience in this area, and can be counted on to help come up with ideas and implementations that will work for everyone.

5: Jihad

A Jihad is a road trip, in which an evangelist visits a large number of ISVs one-on-one to convince them to take some specific action. The classic Jihad is one focused on getting Tier A ISVs to commit to supporting a given technology by signing the technology's Letter of Agreement (LOA – see above).

A Jihad focuses on the Travelling Salesman aspect of evangelism. As in sales, the purpose of the exercise is to close – to get the mark the ISV to sign on the dotted line, in pen, irrevocably. Not to get back to us later, not to talk to the wife about it, not to enter a three-day cooling-off period, but to get the ISV to sign, sign, sign.

If the start of the meeting is the first time the ISV has seen the LOA, then he's not going to sign it at the end of the meeting. Since we're asking for a very serious commitment, we want the ISV to give their signing serious consideration. If the ISV cannot deliver, then his committing to deliver is worse than useless – the ISV's participation may occupy one of a limited number of available slots, keeping some other ISV from participating.

To maximize the chance of getting the ISV to sign during the Jihad visit, make sure that

---

[2] Thanks to Will Gregg.

- The ISV has seen the LOA at least a week before the Jihad visit
- The LOA is very clear about what exactly each side is promising to deliver, and when
- An Officer of the ISV's corporation will be attending the meeting
- Microsoft's Director of DRG has positioned the LOA with sufficient seriousness, in a cover letter or other communication in advance of the meeting
- You make it clear from the start that the purpose of your visit is to answer any questions that they might have, preparatory to *signing* the LOA *while you're there*
- They understand that those who do not sign the LOA, are frozen out of all further information about the technology until it goes into public beta
- They understand (without being crude about it) that you will be making the same offer to their competitors
- You have T-shirts or other swag to give to those who sign. It's amazing what some people will do for a T-shirt.

There are a million tips and tricks to effective road trips, and to being a Road Warrior in general, all of which is beyond the scope of this discussion.

6: FirstWave Program

For the duration of the FirstWave Program, the key evangelism task is bringing the participants' applications' support of the technology to a demonstrable state. This usually requires
- A series of gatherings of the ISVs' key developers in the DevLab, at which the ISVs work on their code with the assistance of PSS and the technology dev team
- Support from PSS, which helps them get trained for the higher volume of calls that they'll get when the technology enters public beta
- Tracking the progress of each ISV's demo application(s) – ideally, on an internal web page – and consolidating the results into regular reports for management (also ideally on an internal web page)
- Orchestrating regular drops of the technology to the ISVs (via a secure web site)
- Planning and starting to execute on the co-marketing benefits promised in the LOA
- Ensuring that bug fix and feature requests from ISVs are prioritized appropriately by the technology team
- Driving and monitoring the development of the evangelical infrastructure (books, courseware, conference sessions, consultants, whitepapers, stacked panels, etc.)
- Laying the groundwork for the technology to be supported by MSDN's one-to-many programs
- Broadening exposure to the technology, within larger ISVs, beyond the targeted demo app to other product teams (be aware of the ISV's internal politics here!)
- Planning and driving the event at which the ISVs' demos will be presented to key industry influencers (press, analysts, ISVs, allies, and competitors).

The launch event should be timed to coincide with the availability of the technology's wide public beta, described below. This timing magnifies the effect of the event: events generate press. Developers read the press and learn that they can download the technology from a given URL – a specific action that they can take *right now*, which will help them start implementing the technology in their own applications *right away*, while it's still fresh in their minds (from the press coverage).

7: Public Beta Release

This is the first beta release for which an NDA is not required. Through the Web and MSDN, such releases can reach millions of developers – far more than Evangelism can possibly work with. At this point, the one-to-many programs must kick in, or Evangelism will be completely swamped and randomized.

The wide beta release is a newsworthy event, and *it must be milk for all it's worth*, because you won't get another such news opportunity until the final version ships. Gather the key industry influencers together at an event – either a stand-alone event, or an event that leverages some larger industry event – and hit 'em

with both barrels, grenades, atomic weapons, and the kitchen sink. Hold nothing back – do everything you can to overwhelm them with how complete, compelling, and inevitable the technology is. Show end-user benefits, developer benefits, the business case, the steak, the sizzle, and everything else – all in a short, punchy, psychologically devastating event. Ideally, everyone attending that event should walk away convinced that the game's over – Microsoft has already won. Have PR work with the press and analysts at and immediately after the launch event, to ensure that the event's coverage is as positive as possible.

Upon the release of the public beta, Evangelism's job changes. It still needs to shepherd the Tier A apps from demo to delivery, but Evangelism also needs to shift gears, to guerilla marketing. One-to-many technology marketing is the job of MSDN, not DRG/Evangelism; let MSDN do it's job. Most of the slides, articles, whitepapers, and other materials developed during the preceding NDA phases, need to be handed over to the appropriate person or group in MSDN, so that they can be widely disseminated, clarified, and hammered repeatedly into the collective consciousness of developers everywhere.

This next phase of evangelism – one of guerilla marketing – I term "the Slog."

8: The Slog

Guerilla marketing is often a long, hard slog.

> **slog** (slǒg) *v.* **slogged, slog•ging, slogs.** –*tr.* To strike with heavy blows, as in boxing.
> –*intr.* 1. To walk with a slow, plodding gait. 2. To work diligently for long hours. –*n.*
> 1. Long, hard work. 2. A long, exhausting march or hike. [Orig. unknown.] –**slog'ger**
> – *American Heritage Dictionary, 1991*

In the Slog, Microsoft dukes it out with the competition. MSDN and Platform marketing are the regular forces, exchanging blows with the enemy mano a mano. Evangelism should avoid formal, frontal assaults, instead focusing its efforts of hit-an-run tactics.

In the Slog, the enemy will counter-attack, trying to subvert your Tier A ISVs to their side, just as you should try to subvert their ISVs to your side. New ISVs should be sought, and directed to MSDN's one-to-many programs. Evangelism should constantly be on the lookout for killer demos, hot young startups, major ISVs, customer testimonials, enemy-alliance-busting defections, and other opportunities to demonstrate momentum for our technology. If bugs are found in our technology, or missing features are found to be critically important, then now is the time to identify and fix them. Stay engaged with the technology development team; ensure that you are a valuable resource for them, not a hectoring pest. Document all of your progress (ideally in regularly updated internal Web pages) and forward it regularly to management. If management is not aware of your progress, your successes, and your stumbling blocks, then they can't help. (They may not help anyway, but they *can't* if they don't know what you need.)

Keep those Tier A ISVs on track to delivery! They are your strongest weapons, and cannot be forgotten.

The elements of the evangelical infrastructure – conference presentations, courses, seminars, books, magazine articles, whitepapers, etc. – should start hitting the street at the start of the Slog. They should be so numerous as to push all other books off the shelf, courses out of catalogs, and presentations off the stage.

Working behind the scenes to orchestrate "independent" praise of our technology, and damnation of the enemy's, is a key evangelism function during the Slog. "Independent" analysts' report should be issued, praising your technology and damning the competitors (or ignoring them). "Independent" consultants should write columns and articles, give conference presentations and moderate stacked panels, all on our behalf (and setting them up as experts in the new technology, available for just $200/hour). "Independent" academic sources should be cultivated and quoted (and research money granted). "Independent" courseware providers should start profiting from their early involvement in our technology. Every possible source of leverage should be sought and turned to our advantage.

I have mentioned before the "stacked panel." Panel discussions naturally favor alliances of relatively weak partners – our usual opposition. For example, an "unbiaed" panel on OLE vs. OpenDoc would contain representatives of the backers of OLE (Microsoft) and the backers of OpenDoc (Apple, IBM, Novell, WordPerfect, OMG, etc.). Thus, we find ourselves outnumbered in almost every "naturally occurring" panel debate.

A stacked panel, on the other hand, is like a stacked deck: it is packed with people who, on the face of things, should be neutral, but who are in fact strong supporters of our technology. The key to stacking a panel is being able to choose the moderator. Most conference organizers allow the moderator to select the panel, so if you can pick the moderator, you win. Since you can't expect representatives of our competitors to speak on your behalf, you have to get the moderator to agree to having only "independent ISVs" on the panel. No one from Microsoft or any other formal backer of the competing technologies would be allowed – just ISVs who have to use this stuff in the "real world." Sounds marvelously independent doesn't it? In fact, it allows us to stack the panel with ISVs that back our cause. Thus, the "independent" panel ends up telling the audience that our technology beats the others hands down. Get the press to cover this panel, and you've got a major win on your hands.

Finding a moderator is key to setting up a stacked panel. The best sources of pliable moderators are: •
- Analysts: Analysts sell out – that's their business model. But they are very concerned that they never look like they are selling out, so that makes them very prickly to work with.
- Consultants: These guys are your best bets as moderators. Get a well-known consultant on your side early, but don't let him publish anything blatantly pro-Microsoft. Then, get him to propose himself to the conference organizers as a moderator, whenever a panel opportunity comes up. Since he's well-known, but apparently independent, he'll be accepted – one less thing for the constantly-overworked conference organizer to worry about, right?

Gathering intelligence on enemy activities is critical to the success of the Slog. We need to know who their allies are and what differences exist between them and their allies (there are always sources of tension between allies), so that we can find ways to split 'em apart. Reading the trade press, lurking on newsgroups, attending conferences, and (above all) *talking to ISVs* is essential to gathering this intelligence.

This is a very tough phase of evangelism. You'll be pulled in every direction at once, randomized by short-term opportunities and action items, nagged by your Tier A ISVs and pestered by every other ISV that wants to become a Tier A. Management will want to know *right now* how you're going to respond to some bogus announcement by some random ISV. Some PM over in Consumer will demand that you drop everything to go talk to an ISV in Outer Mongolia, that's run by an old college chum of his. Competitors will make surprise announcements, lie through their teeth, and generally try to screw you just as hard as you are trying to screw them.

Of course, if you are very, very lucky, there will be no competitors to your technology. But this is almost never the case. ODBC had its IDAPI, OLE had its OpenDoc, COM had its SOM, DCOM has its CORBA, MAPI had its VIM, etc., etc., etc. The existence of a Microsoft technology nearly guarantees that a competitive technology will spring into existence overnight, backed by an impromptu association of Microsoft competitors which have decided to draw yet another Line in the Sand ("If we don't stop Microsoft here, then they are going to take over the whole world!").

Without a competing technology to fight, you just hand everything over to MSDN, give your Tier A ISVs to PSS, and find a new technology to evangelize. But that takes most of the fun out of the game. ☺

9: Final Release

Evangelism of a given technology usually ends with the final, shipping release of that technology. One last big press event, with demos, a tradeshow, press releases, etc., is often called for, showcasing the apps that are sim-shipping and the customers that are using them. In the face of strong competition, Evangelism's

focus may shift immediately to the next version of the same technology, however. Indeed, Phase 1 (Evangelism Starts) for version X+1 may start as soon as this Final Release of version X.

10: *Critical Mass*

The Slog may continue beyond the Final Release, for many months, until Critical Mass is reached. It is possible that Critical Mass will not be reached at all for Version X of a technology, such that Phases 1-9 will have to be repeated – possibly more than once – before ever reaching Critical Mass.

Critical Mass is reached when the technology starts evangelizing itself. When reviews subtract points if it's not supported; when analysts say "great product plan, but what about [Technology Name]?"; when VC's won't fund a company unless it supports [Technology Name] – that's Critical Mass. At that point, Evangelism of the technology stops, and Evangelism's resources are applied to other technologies – or, if you're lucky, moves into the Mopping Up phase.

11: *Mopping Up*

Mopping Up can be a lot of fun. In the Mopping Up phase, Evangelism's goal is to put the final nail into the competing technology's coffin, and bury it in the burning depths of the earth. Ideally, use of the competing technology becomes associated with mental deficiency, as in, "he believes in Santa Claus, the Easter Bunny, and OS/2." Just keep rubbing it in, via the press, analysts, newsgroups, whatever. Make the complete failure of the competition's technology part of the mythology of the computer industry. We want to place selection pressure on those companies and individuals that show a genetic weakness for competitors' technologies, to make the industry increasingly resistant to such unhealthy strains, over time.

12: *Victory*

Some technologies continue as competitors long after they are true threats – look at OS/2, the Operating System that Refused to Die. It is always possible – however unlikely – that competitors like OpenDoc, SOM, OS/2, etc., could rise from the dead...so long as there is still development work being done on them. Therefore, final victory is reached only when the competing technology's development team is disbanded, its offices reassigned, its marketing people promoted, etc. You have truly and finally won, when they come to interview for work at Microsoft.

Victory is sweet. Savor it. Then, find a new technology to evangelize – and get back to work! ☺

Strategic Plan

Apple Computer, WLM, Windows NT, and the Mac OS

James Plamondon
September 21, 1995

Objective

The objective of this strategy it to convert Apple Computer, Inc., from OS foe to hardware friend.

Executive Summary

This strategy involves two coordinated efforts:

1. *WLM:* We make WLM the most compelling solution for the deployment of Mac OS-based applications. If successful, this effort will establish Win32 as the *de facto* API of the Mac OS.

2. *Windows NT/PowerPC:* We make Windows NT the most compelling operating system for PowerPC-based servers and workstations – specifically, for high-end graphics and publishing applications. If successful, this effort will allow Windows NT to usurp the Mac OS on Apple's own high-end hardware.

Pros and Cons

Positive result: Most cross-platform applications will be written first for Windows, and work best on Windows.

Negative result: Many apps that might otherwise have been Windows-only, will be ported to the Mac using WLM – but this strengthens Win32 on the Mac, and thus the positive result.

WLM

With its new support for the PowerMac and OLE, WLM is poised to become a major success on the Macintosh. An independent consultant who previously recommended VC/Mac 2.0 to only 20% of his clients, now recommends VC/Mac 4.0 to 90% of them – and has told Apple that he expects 50% of the units sold on the Mac in 12 months to be using WLM. Apple itself is paying a consultant to port PeopleSoft to the Mac using WLM.

Unable to offer developers a cross-platform development alternative to WLM, Apple wants to make sure that WLM-based applications take the best possible advantage of unique Mac OS features (such as Apple Guide, PowerTalk, QuickDraw GX, etc.). Apple hopes that this will prevent the Mac from becoming awash with WLM-based apps that treat the Mac as a least-common-denominator platform, which would just speed the demise of the Mac OS.

We don't care whether Apple succeeds in its attempt or not. Their endorsement of the Win32 API on the Mac effectively ensures that ISVs write their apps first for Windows; that the resulting apps work best on Windows; and that the Mac OS is the recipient of second-class, ported apps.

While the availability of WLM will cause many apps that might otherwise have stayed Windows-only to support the Mac too, each of these ported apps strengthens the WLM standard – which in turn increases the attractiveness of writing to Win32, and hence writing first and best for Windows.

Our efforts with WLM are only one flank of a double-envelopement maneuver – the other flank being Windows NT on the PowerPC (see below). Windows NT/PowerPC comes at the Mac OS from the high end down; WLM comes at the Mac OS from the low end up. Therefore, VC/Mac and WLM should not focus solely on the Mac's high end; they must continue to target lower-end systems, to. Specifically, VC/Mac should continue to support the 68K Mac.

Tactics:

- Apple: Get Apple to publicly endorse WLM; to use it in their own cross-platform development; to assist in its development; to evangelize its use by ISVs; to make Copland like WLM; to include WLM in Copland.

- DAD/CSD: Use WLM in our own apps, especially those that are likely to be bundled with Macs, given away free, or otherwise have high penetration; ship as few versions of WLM as possible; ship the WLM DLL file intact and unaltered; drop the licensing restrictions on WLM.

- PSD/BSD: Help keep WLM in sync with additions to Win32; work with Apple to make NT/SFM an even better player in Mac networks; do what is necessary (technically and marketing) to take the graphics and publishing business away from the Mac.

- DD: Work closely with Apple and Microsoft's internal groups to support WLM; license WLM directly to Mac tool vendors (after or instead of bundling); keep up with additions to Win32; continue support for 68K.

Windows NT on the PowerPC

As Apple's PowerMacs converge with IBM and Motorola's PReP machines onto the CHRP standard, we have a golden opportunity to set Apple's strategy on its head, and usurp the Mac OS on Apple's own hardware.[3]

CHRP boxes from the current PReP vendors will run Windows NT very well; NT is their focus. So Apple will be faced with direct competitors that offer equivalent hardware, running Windows NT instead of the Mac OS – and Windows NT apps instead of Mac OS apps. If these Windows NT/CHRP applications have a price/performance advantage over the equivalent Mac OS/CHRP apps, Apple's going to be in a very tough competitive position.

And given that the Mac OS won't support threads (let alone symmetric multi-processing) for over a year, it seems likely that – all else being equal – Apple is indeed likely to be in a tough spot, at the high end, at least. SUR's lack of Plug-n-Play, and steep hardware requirements, will keep it off of the desktop in the home and education markets, leaving those to the Mac OS for a while longer.

At the high end, though – workstation and server apps – NT/PowerPC market is about to explode, due to the confluence of three factors:

1. SUR: The Shell Update Release is going to make Windows NT more palatable to everyone – especially those who might otherwise choose a Macintosh. The wave of PR/Marketing surrounding its release, while being nothing like the Windows 95 hype (I assume), will do much to increase the acceptance of NT as a desktop system. And the more rapidly Windows NT takes off, on any chip, the better for Windows NT on other chips.

2. VC/NT/PowerPC: The release of Visual C++ 4.0 for Windows NT/PowerPC, at year's end, will make it easy for the developers of Windows 95 applications to meet the demand for the PowerPC-native versions that SUR will generate.

3. CHRP: Apple adopts a hardware standard that supports Windows NT, at just exactly the same moment that Windows NT's user interface is improved beyond that of the Mac OS. (You gotta love it.) Apple (and its licensees), IBM, Motorola, and the Taiwanese, collectively ships millions of Windows NT-capable boxes.

To make it possible for a mixed shop to standardize on Windows across the board, we'll need a version of Windows NT that can run on Apple's installed base of PowerSurge Macs. Motorola says that this work is not theoretically difficult, although they are concentrating their attentions on CHRP first, which is fine.

We also want to lower the cost of switching from the Mac OS to Windows NT/PowerPC, given hardware that can run both OSs; this can be done (in part) be encouraging ISVs (including Microsoft's apps groups) to include the NT/PowerPC version of an app on the PowerMac SKU of that same app (as well as on Windows SKU).

Tactics:

- Apple: Work with Apple to ship NTS on Shiner; to improve the support for the Mac OS in NT/SFM.

- Mac OS Licensees: Work with the licensees to ensure that their CHRP and PowerSurge machines run Windows NT really well (dual-boot).

---

[3] Doing the same to IBM's OS/2 on PowerPC strategy at the same time – slaying two birds with one stone.

- DAD/CSD: Produce complete, optimized NT/PowerPC versions of every app for which you currently plan to ship either a Mac OS or an NT/Intel version, including support for DAO, Jet, etc.; ship the NT/PowerPC version on both the Windows CD SKU and the PowerMac CD SKU. (Except for really low-end home/education Mac OS stuff.)

- PSD/BSD: Work with Apple to make NT/SFM an even better player in Mac networks; do what is necessary (technically and marketing) to take the graphics and publishing business away from the Mac; produce Intel-peer, complete and optimized versions of NT, BackOffice, etc.

- DD: Produce Intel-peer, complete and optimized versions of our entire tools suite, including DAO, Jet, etc., keeping in sync with all Intel releases, etc.

Appendix A: Windows Layer for the Macintosh (WLM)

WLM is Microsoft's implementation of the Win32 API on top of the Mac OS. A developer can take a Win32-based application's code, recompile it, and link it to WLM, with the result being an application that looks and feels like a native Macintosh application – that was written entirely to the Win32 API.

The wide adoption of WLM is good for Microsoft, because it ensures that applications are written first for Windows, and work best on Windows, with the Mac as a second-class platform.

WLM is currently available only as an element of Microsoft Visual C++ Cross-Development Edition for the Macintosh, but it will also be available soon in a separate WLM/MFC SKU. The success of the WLM SKU is nearly assured, as its creation was driven by the desires of the Mac compiler vendors (Metrowerks and Symantec).

WLM's strengths were obscured by its lack of support for OLE and, especially, the PowerMac, in its first release (in VC/Mac 2.0, November, 1994). These features are being added to the forthcoming version, to be released in VC/Mac 4.0 (November, 1995).

WLM 4.0 still lacks support for multimedia and OLE Controls, which are currently being investigated for support in next year's release.

Appendix B: The PowerPC, PReP, CHRP, and Copland

The PowerPC is a RISC-based CPU manufactured by IBM and Motorola.[4]

There are two different standards for personal computers based on the PowerPC CPU: the PowerPC Reference Platform (PReP), and Apple's PowerMac.

All of Apple's high-end Macs, and an increasing share of its mid- and low- end Macs, are based on the PowerPC. All PowerPC-based Macs are referred to as "PowerMacs." Early PowerMacs were hard-wired to be big-endian (as required by the Mac OS), and thus could not run Windows NT (which requires a little-endian architecture). Older PowerMacs also used the NuBus bus architecture. The newest PowerMacs, however, are endian-switchable at run-time, and use the PCI bus, so that they could conceivably run Windows NT in addition to the Mac OS. These newest PowerMacs – the 8500 and 7500, code-named the PowerSurge line – can be viewed as a half-step by Apple towards CHRP-compliance (see below).

Apple has licensed its PowerMac hardware designs (including PowerSurge) and the Mac OS itself to three small companies (Power Computing, DayStar, and Radius), which are now shipping Mac compatibles. These Mac OS licensees are very interested in offering Windows NT on their future CHRP systems, both to hedge their bets against the potential failure of the Mac OS, and to differentiate their Mac compatibles from Apple-logoed Macs.

IBM, Motorola, FirePower, and others, manufacture PowerPC computers that are based on the PowerPC Reference Platform (PReP) standard, defined by IBM and Motorola. All ship with Windows NT pre-installed.

The Common Hardware Reference Platform (CHRP) is the unification of the PowerMac and PReP designs, into a single hardware standard that is intended to run Windows NT, the Mac OS, and others (OS/2, Solaris, AIX, NetWare, NextStep, etc.). CHRP is based on standard industry components – for everything except the core PowerPC CPU. CHRP's backers hope to make it a strong second standard, challenging the Intel chip/chipset/motherboard monopoly. CHRP is also backed strongly by the Taiwanese New PC Consortium (TNPC), whose board-manufacturing businesses have recently been idled by Intel's entry into that market.

The CHRP spec is expected to become final late in 1995. Motorola has Windows NT up and running on a CHRP prototype today, which they plan to demonstrate at Fall Comdex in November 1995. CHRP systems running Windows NT are expected to hit the market as soon as Q1 96.

Apple has stated that it will ship its first CHRP-compliant computers in late 1996, but much of their tardiness is due to the expectation that these machines will be running the Mac OS, which is not expected to run on CHRP-compliant systems until that time. Apple may be able to ship a CHRP box that runs Windows NT well before it can ship one running the Mac OS (just as IBM is now shipping NT on its PReP computers, while OS/2 for PowerPC languishes in development).

"Copland" is the code-name for the next version of the Mac OS, a.k.a. "System 8," designed specifically to take advantage of the CHRP standard. Apple is saying publicly that Copland will be ready in late 1996, in time for their first CHRP systems – but it is widely believed that Copland will not be released until mid-to-late 1997. It is known that Apple is porting their existing Mac OS – System 7.5 – to CHRP, in case Copland does not ship on schedule.

---

[4] Versions 601, 603, and 604 are shipping now; the 620 is under development; the project to build the 615 – rumored to have 486 compatibility in silicon – has apparently been scrapped.

Appendix C: Apple and Windows NT

The Mac OS is a crummy OS for servers, and Apple knows it. On the other hand, Windows NT is doing very well on servers, and its support for Mac file and print sharing[5] makes it a natural server for Mac-centric environments.

At one point in late 1994, Apple's Director fof CHRP Marketing, Jim Gable, was running a skunk-works project with Motorola, with the intention of implementing Windows NT on its PowerSurge-based "Shiner" server. When Spindler heard about this skunkworks effort early this year, he went ballistic, and shut it down.

Since then, Apple has reorged, and Don Strickland – formerly of Mac OS Licensing – has taken over as VP of Apple's enterprise marketing. He is backing a new effort to get NT as the default OS on "Shiner" (which has been redesigned to be fully CHRP-compliant), and has secured the approval of Apple's President and CEO, Michael Spindler. Microsoft, Motorola, and Apple met to discuss this new effort the week of September 25th, and agreed that it could be ready to ship in Q1 96. The shipment of an Apple-logoed server with Windows NT on it. will be opposed by Apple's Mac OS group, and they may succeed in killing the project – but Don Strickland's backing will probably help Shiner ship with NT on board.

---

[5] Windows NT Server has a collection of "Services for Macintosh" that make it, as Stewart Alsop said in InfoWorld, "The obvious server for Macintosh networks."

Appendix D: Apple's Down – But Not Out

Declining ISV support. Almost 90% of Mac ISVs are also writing for Windows. Apple is having pay Windows ISVs such as PeopleSoft to produce Mac versions of their applications – and even then, PeopleSoft is using WLM, rather than writing native Mac OS code. Metrowerks, the leading Mac tool vendor, recently released a Mac-hosted, Windows-targeting cross-compiler – to a standing ovation – that will accelerate this migration to Win32, and they are in negotiations to license WLM as well. Adobe recently blamed its poor financial results on its over-dependence on the Mac, and is moving drmatically towards Windows and Windows NT. OpenDoc—a key element of Apple's system software strategy—is rapidly losing mindshare; its failure will seriously discredit the Mac OS group within Apple.

Declining Market Share. The Mac OS's share has dropped to having under 8% of the OS market, and analysts project it to continue to spiral downwards. SPA recently reported that sales of Mac applications were down 7% in the second quarter of '95, compared to '94. The increasing sales of Windows 95 will push the Mac OS' market share even lower by early next year.

Declining Profitability. Meanwhile, Apple has announced a new line of its PowerPC-based PowerMacs – the PowerSurge line – that offer all the power of their previous configurations, at much lower cost. Sales of their older high-end Macs have slowed, awaiting the new PowerSurge Macs – which are unavailable, due to part shortages. As a result, Apple recently announced a dramatic 48% decline in profitability.[6]

Declining Technical Advantage. Because Windows 95 has reached technical parity with the Mac OS (roughly speaking), Apple can no longer charge a premium for the Mac – it has to be price-competitive with IBM, Compaq, and the other first-tier Wintel vendors. But unlike its PC competitors, Apple has a huge software R&D budget, devoted to maintaining and advancing its proprietary OS. The next version of the Mac OS is expected to be released in 1997 – 18 months or more away. Even then, the new version will not offer features dramatically superior to Windows 95, let alone Windows NT. And Windows will, by then, have extended its lead even further – in quality, in installed base, and in application support. Finally, a forthcoming report from IDC shows that users are more productive using the Windows 95 UI than the Mac UI – robbing the Mac OS of its one last claim to superiority.

Entrenched Base. Apple does have an installed base of around 16 million active users, mostly in the home, education, and in graphics & publishing. The home and education units are going to remain in place well into the next millenium; those markets just don't turn over their hardware quickly – so there will be a continuing (albeit declining) demand for Mac OS software for years to come. On the other hand, the graphics and publishing market turns over its hardware very quickly, since they are usually riding the very leading edge of the power curve – so Windows NT Workstation should be able to blow the Mac OS off of Apple's own hardware in this market, in pretty short order.

Apple's Spring Offensive. Despite its many reverses, Apple is not yet dead – nor is the Mac OS. Apple has reorganized its developer support efforts, and can now evangelize and support its few remaining developers efficiently. When a developer release of Copland becomes available early next year, Apple will have something tangible to evangelize, giving the Mac OS lots of new mindshare. And OpenDoc will finally be released in final form, to great fanfare. The impending availability of CHRP-compliant Macs will also encourage a new crop of Mac OS licensees to sprout at about this time.

---

[6] The "Osborne Effect" in action. When Adam Osborne enthusiastically described the features of his forthcoming *Osborne II* computer, sales of the existing *Osborne I* fell so sharply that his company went bankrupt before the new model could be brought to market.

In short, we should expect to see a major Spring Offensive from Apple in 1996, which we will need to blunt and turn back.

LETTER OF AGREEMENT (LOA)

BETWEEN MICROSOFT CORPORATION ("MS")
AND _____ *("COMPANY")*

DATE_____

Microsoft® Developer Relations Group Internet Program
for Internet Authoring Tools

This LOA sets out the agreement between MS and COMPANY with respect to the development of authoring tools using MS Internet client and server platform technology as described below. The LOA, in conjunction with attached material, describes the MS Developer Relations Group Internet program for Internet authoring tools, its benefits, and requirements for participation through September 1997, or the release of Internet Explorer 4.0, whichever is later.

In brief, this program provides COMPANY with technical assistance in implementing exciting new features in COMPANY's Internet authoring tools , as well as co-marketing assistance in promoting its Internet authoring tools. The benefit to MS is in having Internet authoring tools for creating compelling, exciting products, which will showcase new features of MS's Internet solutions.

Program Benefits — Technical Assistance from MS

1. MS will provide premier-level product support to COMPANY for the Internet technologies named in this document, free of charge.

2. MS will provide COMPANY access to the latest releases (including interim builds) of Internet related platforms, products and development tools, in accordance with their respective beta testing agreements, through a private web site prior to any broad software release.

3. MS will provide a special "technical envoy" from MS's Developer Relations Group to assist in clearing roadblocks, which might arise during the development of COMPANY's Internet authoring tools, including scheduling meetings between the COMPANY and Microsoft at Microsoft's discretion and subject to time constraints, schedules and availability.

4. MS will invite COMPANY for special implementation lab weeks and workshops in Redmond, providing COMPANY great support for developing and debugging its Internet authoring tools. Travel and lodging expenses are COMPANY's responsibility.

Program Benefits — Co-marketing Assistance from MS

1. MS will provide information on COMPANY's Internet authoring tools and their support for the features described in "Requirements for Participation" (below), or demonstrate COMPANY's Internet authoring tools to leading Independent Content Providers (ICPs) participating in the Developer Relations Group Internet Program.

2. MS will invite COMPANY to exhibit in a Microsoft-affiliated area at targeted conferences, in accordance with the terms of the applicable Exhibitor Agreement. Possible conferences include Comdex Canada (Toronto, Ontario, July 9-11), Internet World/Summer (Chicago, IL, July 21-25) and Microsoft's Professional Developers Conference (San Diego, CD, September 23-26). All conferences are tentative and subject change.

3. MS will invite COMPANY to propose a demonstration of its Internet authoring tools to be included in a live satellite broadcast ("Microsoft at the Movies") focused on Internet Explorer 4.0, possibly in conjunction with the Microsoft Professional Developers Conference.

4. MS will invite COMPANY to include a demonstration version of COMPANY's Internet application tools on a CD-ROM, if such a CD-ROM is produced. Over 12,000 attendees received CD-ROMs at the most recent comparable event.

5. COMPANY's Internet authoring tool will have the opportunity to be featured on Microsoft's Site Builder Network. Terms and conditions of the agreement for featuring COMPANY's Internet authoring tool have not be finalized and are subject to Site Builder Network approval.

6. COMPANY's Internet authoring tool will be featured for one or more weeks in the new "Tool of the Week" section of Microsoft's Tools Gallery. (currently located at http://www.microsoft.com/gallery/files/tools/) Within 60 days following the availability of COMPANY's pre-release or released Internet authoring tool complying with "Requirements for Participation" (below).

7. COMPANY will have the opportunity to license selected content from MS's Site Builder Network Galleries, for inclusion with COMPANY's Internet authoring tools. Terms and conditions of this license have not been finalized.

8. MS will make best efforts to support any press releases or other public activity that COMPANY carries out relating to the Internet and the use of MS technology.

9. MS may provide other co-marketing avenues from time to time, such as site contests and joint advertising. By being on this program, COMPANY will be among the first to be invited to participate in these co-marketing initiatives.

Requirements for Participation
Product Requirements

1. COMPANY will obtain the "Designed for Windows NT and Windows 95 Logo" for its Windows-based Internet authoring tools. If COMPANY's Internet authoring tool is end-user focused COMPANY will obtain the "Office97 Compatible Logo", additionally, if COMPANY's Internet authoring tool adds value to the Microsoft BackOffice Suite, COMPANY will obtain the "Designed for Microsoft BackOffice Logo". Information regarding Microsoft's Logo programs is available at http://www.microsoft.com/windows/thirdparty/ and http://www.microsoft.com/office/compatible/. Microsoft will consider paying the expenses of obtaining any logos for the COMPANY's Internet authoring tools, on a case-by-case basis, depending on the level of technology support offered by COMPANY's Internet authoring tools.

2. COMPANY's Internet authoring tools will check inserted ActiveX controls and Java applets for digital certificates using the WinVerifyTrust interface, there by supporting Microsoft's Authenticode technology. For ActiveX controls and Java applets without digital certificates, or whose certificates are invalid, (out of date, revoked) COMPANY's Internet authoring tools will provide appropriate warnings. (See http://www.microsoft.com/msdn/sdk/platforms/doc/sdk/win32/func/src/f91_32.htm for information on the WinVerifyTrust interface.) Additionally COMPANY's Internet authoring tools will use the IClassFactory2 interfaces to enforce ActiveX control licensing. (See http://www.microsoft.com/msdn/sdk/platforms/doc/activex/src/comext_8.htm for information on the IClassFactory2 interfaces.)

3. COMPANY will enhance its Internet authoring tools to host Design Time Controls, as described in the Web Design-Time Control SDK, available as http://www.microsoft.com/intdev/sdk/dtctrl/dcsdk.exe. Microsoft is developing a selection of Design Time Controls, and is considering licensing those controls for redistribution with 3rd party authoring tools. Possible controls include, data binding controls, CDF creation controls,

Visual CSS positioning controls, and Multimedia Effects controls. The specific controls and licensing terms and conditions have not been finalized.

4. COMPANY will enhance its Internet authoring tools to support Cascading Style Sheets (CSS) as defined in the Internet Client SDK at http://www.microsoft.com/sitebuilder/features/inetsdk.asp and by the W3C at http://www.w3.org/pub/WWW/Style/. This support should include support for all CSS elements and attributes, and the ability for authors to visually change any element attributes.

5. COMPANY will enhance its Internet authoring tools to support CSS positioning as defined in the Internet Client SDK at http://www.microsoft.com/sitebuilder/features/inetsdk.asp and by the W3C at http://www.w3.org/pub/WWW/TR/WD-positioning. This support should include the ability for authors to position any element with CSS defined precision.

6. COMPANY will enhance its Internet authoring tools to support Internet Explorer 4.0 multimedia effects, as defined in the Internet Client SDK at http://www.microsoft.com/sitebuilder/features/inetsdk.asp. This functionality should allow authors to define interactive, media rich web pages, by providing an easy interface for using effects such as transitions, sequences, paths, audio, complex graphics, spites, and the other effects provided with Internet Explorer.

7. COMPANY will enhance its Internet authoring tools to support the creation of Channel Definition Format (CDF) files as defined in the Internet Client SDK at http://www.microsoft.com/sitebuilder/features/inetsdk.asp and by the W3C at http://www.w3.org/pub/WWW/Style/. This support should include the ability for authors to select any content, including cross-site content, and then automatically generate the CDF file, and required META tag information.

8. COMPANY will enhance its Internet authoring tools to support Internet Explorer 4.0 data binding attributes, as defined in the Internet Client SDK at http://www.microsoft.com/sitebuilder/features/inetsdk.asp. Internet Explorer 4.0 data binding allows authors to asynchronously push data to specified HTML pages. This functionality should allow authors to select elements to bind to available data sources. Allowing authors to define the format of returned data, i.e. repeated table, or current row.

9. COMPANY will enhance its Internet authoring tools to support the PICS rating system, as defined by the W3C at http://www.w3.org/pub/WWW/TR/REC-PICS-services-961031.html. This support should include the ability to visual set the PICS settings, and then generate and insert the required META tag information.

10. COMPANY's Internet authoring tools will support Active Server Pages (ASP). This support must include ASP shorthand syntax "<%...%>" and value added support for the server components that ship with IIS 3.0 or later. (See http://www.microsoft.com/iis/ for information regarding Active Server Pages and IIS components.)

11. COMPANY will enhance its Internet authoring tools to add Microsoft Transaction Server (MTS) support to all server side components that ship with its Internet authoring tools, including Java applets. (See http://www.microsoft.com/transaction/ for information regarding Microsoft Transaction Server.)

12. COMPANY will enhance its Internet authoring tools to add value-added support to Microsoft's SiteServer and other server side components. (See http://www.microsoft.com/siteserver/ for information regarding Microsoft's SiteServer.)

13. COMPANY's Internet authoring tool will host the Microsoft Internet Control (the Microsoft Internet Explorer HTML: engine, also known as "WebBrowser") as its integrated browser, there-by providing authors with integrated previewing capability. (See http://www.microsoft.com/workshop/prog/sdk/docs/iexplore/webrowse.htm for information on integrating the WebBrowser control.) Additionally COMPANY will license Internet Explorer 4.0 or later for redistribution with their Internet authoring tools. (See http://www.microsoft.com/ie/ieak/ for information on redistributing Internet Explorer.

14. COMPANY's Internet authoring tools will host the Microsoft Active scripting interfaces, allowing interactive authoring and debugging of scripts and Java applets.

15. COMPANY will enhance some Java components and applets that are redistributed with its Internet authoring tools to utilize Microsoft's Application Foundation Classes, including Microsoft's AFC Enterprise classes for some server based Java components. Additionally COMPANY will license and redistribute the AFC classes with its Internet authoring tools.

16. All ActiveX controls and Java components that are distributed with COMPANY's Internet authoring tools will be stored in CAB files and digitally signed using Microsoft's Authenticode™ code signing specification. (See http://www.microsoft.com/workshop/prog/cab/ for information on CAB files and http://www.microsoft.com/workshop/prog/security/authcode/codesign.htm for information on signing CAB files.)

17. COMPANY's Internet authoring tools will support creation of Web Collections, a W3C standard derived from Microsoft's SiteMap technology. Web Collections are used to represent the Table of Contents and Keyword Index files used by the MS HTML Help technology.

18. COMPANY will register their Internet authoring tools with the Microsoft Tools Gallery, currently located at http://www.microsoft.com/gallery/files/tools/.

19. COMPANY will register their Internet authoring tools with the Microsoft J/Advantage program. Information regarding the J/Advantage program can be found at http://www.microsoft.com/java/.

20. COMPANY agrees to display marketing materials (e.g. small signs, monitor toppers) noting product's support for Microsoft's technologies (e.g. Internet Explorer, Microsoft Java VM, AFC) whenever PRODUCT is being demonstrated publicly (e.g. booths at trade shows, conferences, seminars, roadshows.)

21. COMPANY will provide by June 15th, a quote from its most senior executive stating COMPANY's support for the Microsoft Dynamic HTML, another supporting CDF, and a third supporting both the Dynamic HTML and CDF, and the right to use these quotes in Internet-related press releases for 60 days thereafter.

22. COMPANY's Internet authoring tool will support for inserting the authorized Microsoft Internet Explorer logo on a page through an easy to use interface, in accordance with the applicable logo agreements. Additional information can be found at http://www.microsoft.com/ie/logo/.

Site Requirements

1. If COMPANY maintains its own web site, COMPANY will use the Windows NT Internet Information Server 3.0 or later as the primary http server on its site and host at least one page showing support for Active Server Pages.

2. COMPANY will add Microsoft to the list of "Technology Alliances and Partnerships" as listed at http://www.allaire.com/partners/.

3. COMPANY will display the authorized Microsoft Internet Explorer logo on its default home page, in accordance with the applicable logo agreements. Additional information can be found at http://www.microsoft.com/ie/logo/. Or COMPANY agrees not to display any competitive product logos on their home pages.

4. If COMPANY provides their Internet authoring tools for downloading from their web site the setup program will be digitally signed in accordance with Microsoft's Authenticode™ code signing specification. (See http://www.microsoft.com/workshop/prog/security/authcode/codesign.htm for information on signing executable files.) NOTE: Site Builder Network requires all downloadable files to be digitally signed.

Event Requirements

1. COMPANY will use Internet Explorer 4.0 or later as the default browser for all demonstrations and keynotes at Microsoft sponsored events, and will give default to Internet Explorer 4.0 or later but giving equal use to other competing products at all non-Microsoft sponsored events.

Time Frames

COMPANY agrees to meet the following schedules in order to enter and ensure continued participation in this program:

1. COMPANY should have completed requirement 1 listed under the "Site Requirements" section within two weeks of entering into this agreement. COMPANY should have completed all requirements listed under the "Site Requirements" section within one month of entering into this agreement.

2. COMPANY should make a "Beta" release of its Internet authoring tools available for download via its web site within four months of entering into this agreement, meeting all requirements listed under the "Product Requirements" section.

3. COMPANY will ship their Internet authoring tools within three months of the shipment of Internet Explorer 4.0. Please note that participation in some marketing activities may require a shorter and less flexible schedule as well as cross-platform versions of COMPANY's products. Microsoft will use the following COMPANY-supplied information to track company's eligibility (all data will be treated confidentially as described in the "Confidentiality" section):

Product Name(s)	Platform(s) Supported	Product Description(s)	Expected "WWW Beta"	Expected Final Shipment Date
			--	

COMPANY WEB SITE NAME: _____

COMPANY WEB SITE URL: _____

CONTRACTORS USED (if any): _____

Confidentiality

MS and COMPANY shall keep in confidence the existence of this LOA, any of the terms and conditions described herein, and any confidential or proprietary documents or information relating to our products or programs. Neither MS nor COMPANY shall disclose any such information to a third-party except as may otherwise be permitted in the existing Non-Disclosure Agreement between COMPANY and MS, dated

_____.

Accepted and agreed to:
By signing below, you are agreeing:
- to the terms and conditions of the MS Developer Relations Group Internet program for Internet Authoring Tools outlined in this letter
- that you are an officer of your company with authority to enter into this agreement with MS

_____   MICROSOFT®
CORPORATION

By: _____   By: _____

Name: _____   Name:

Title: _____   Title:

Date: _____   Date:

Attachments:
  MS STANDARD NDA

10

11

12

14

15

16

17

18

20

21

23

25

26

27

28

29

30

32
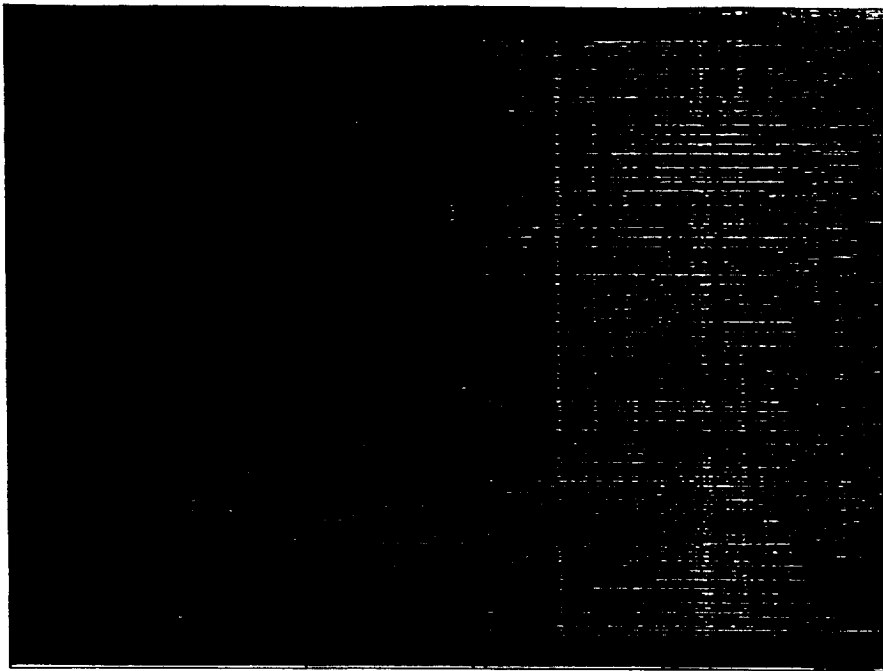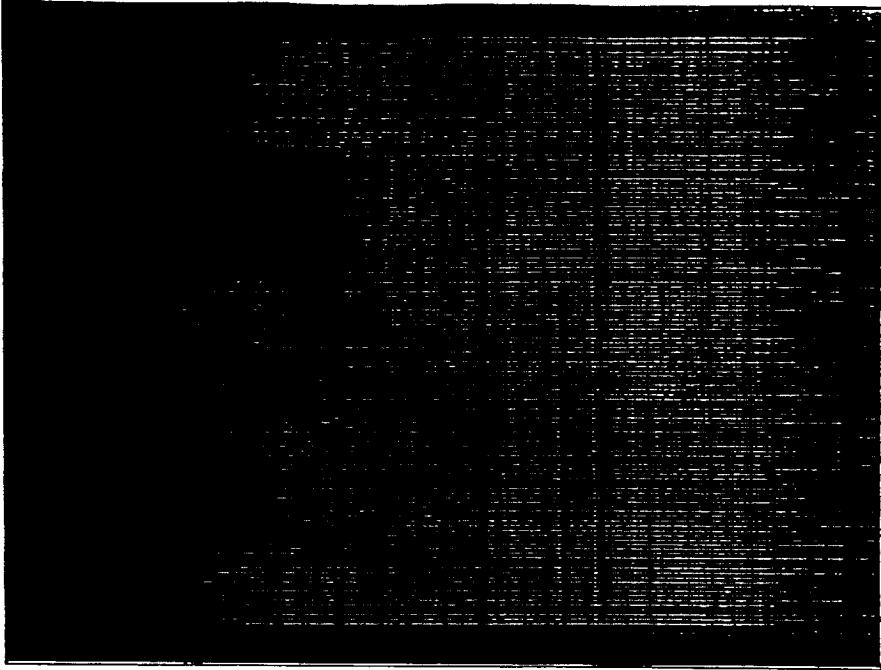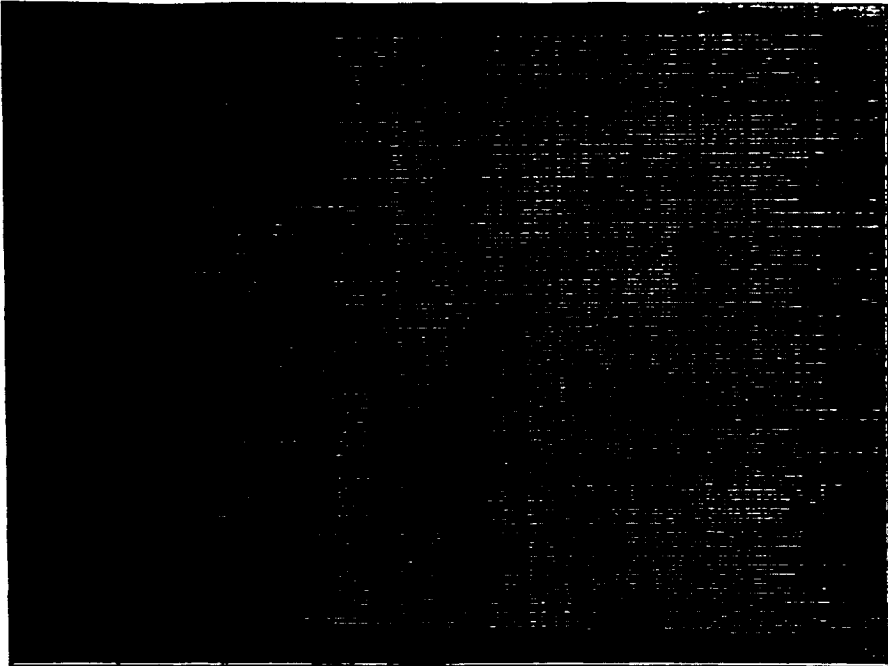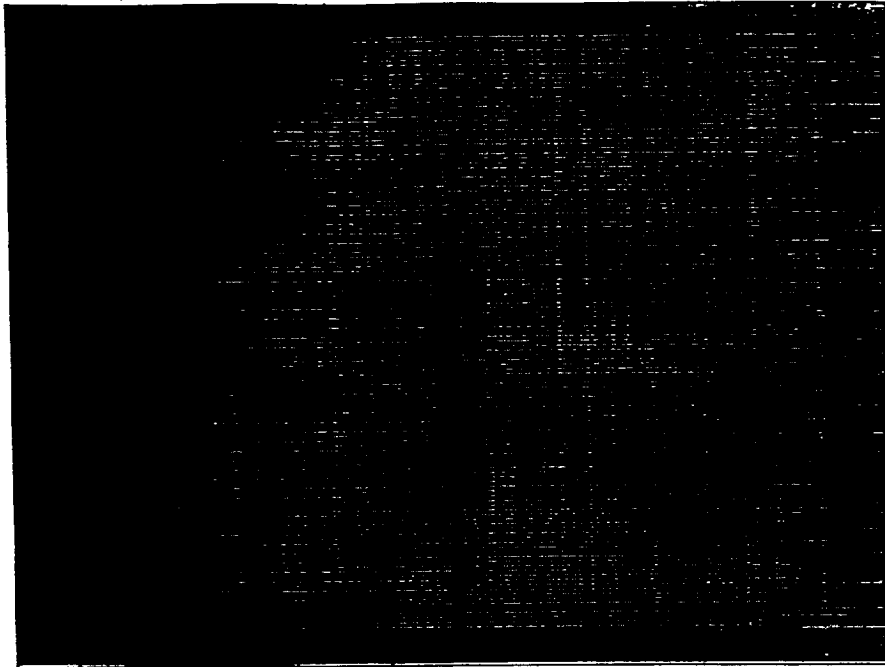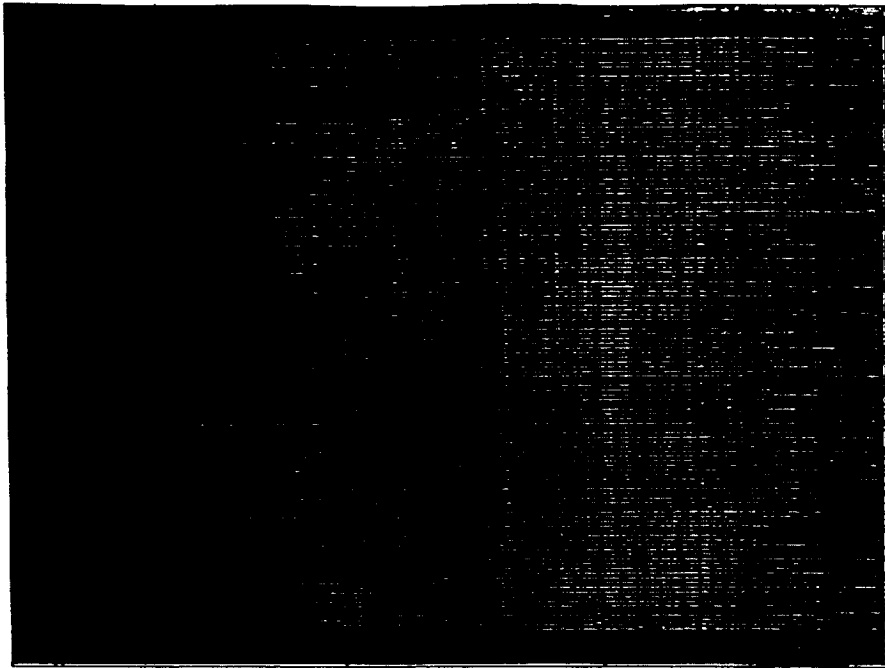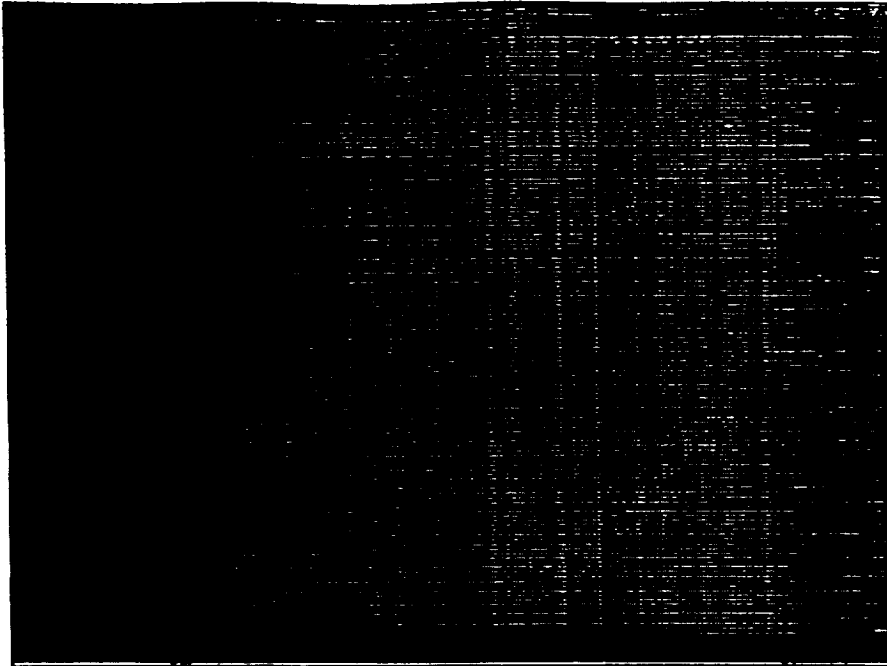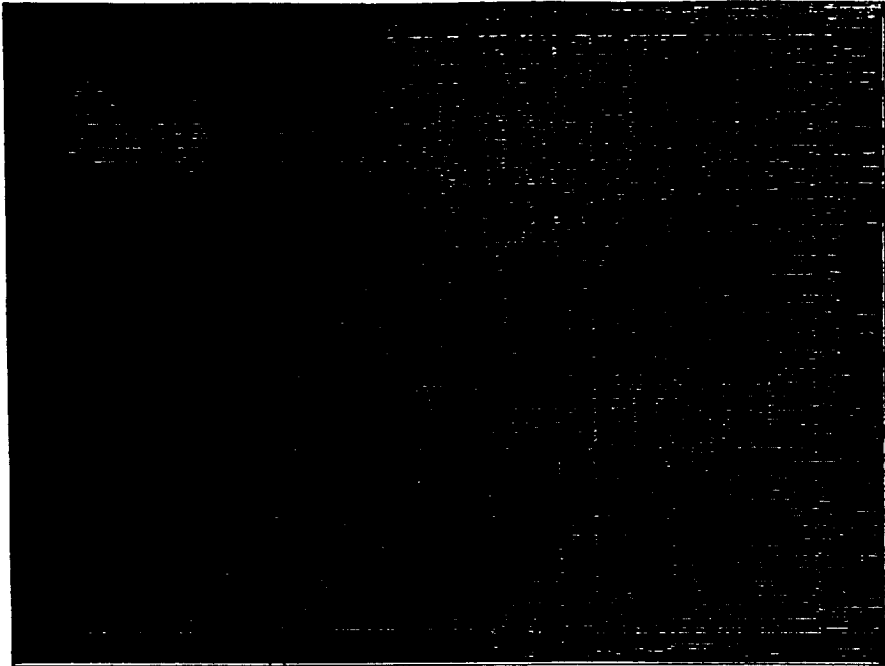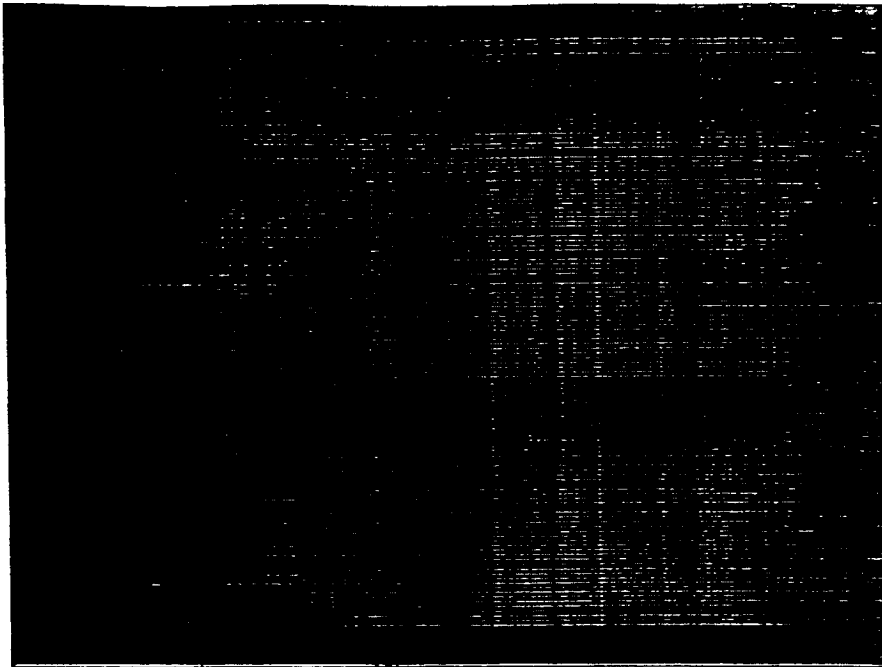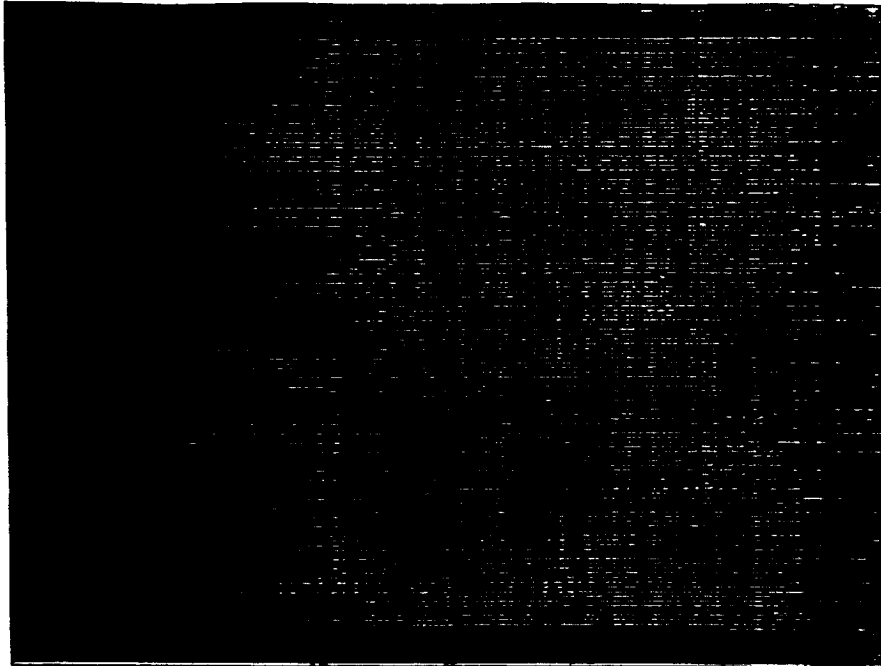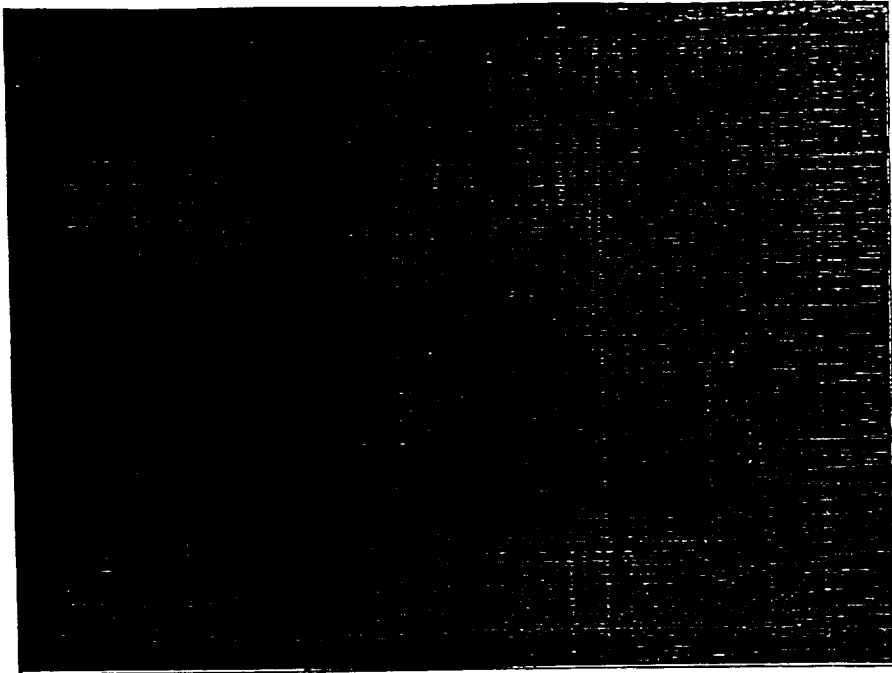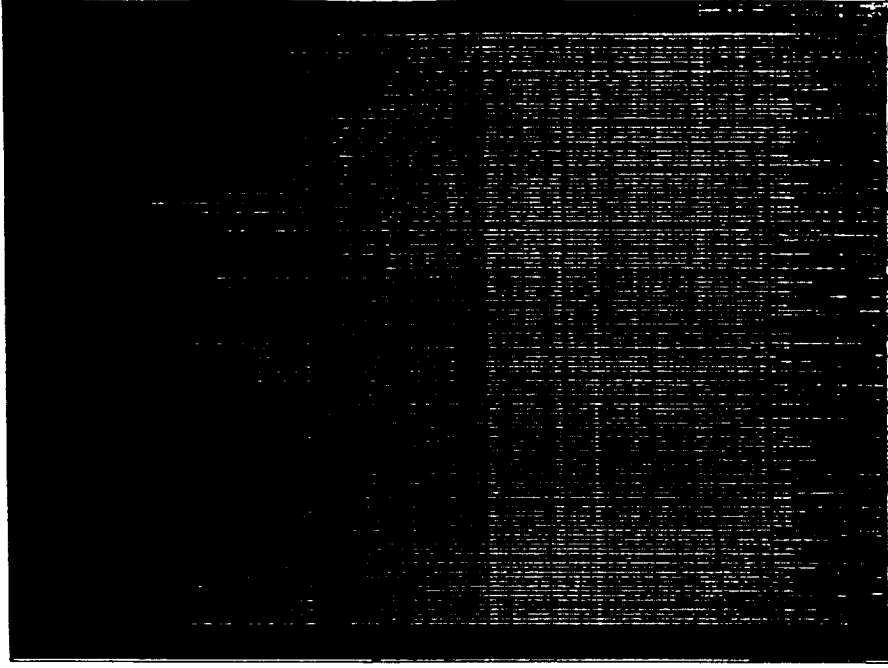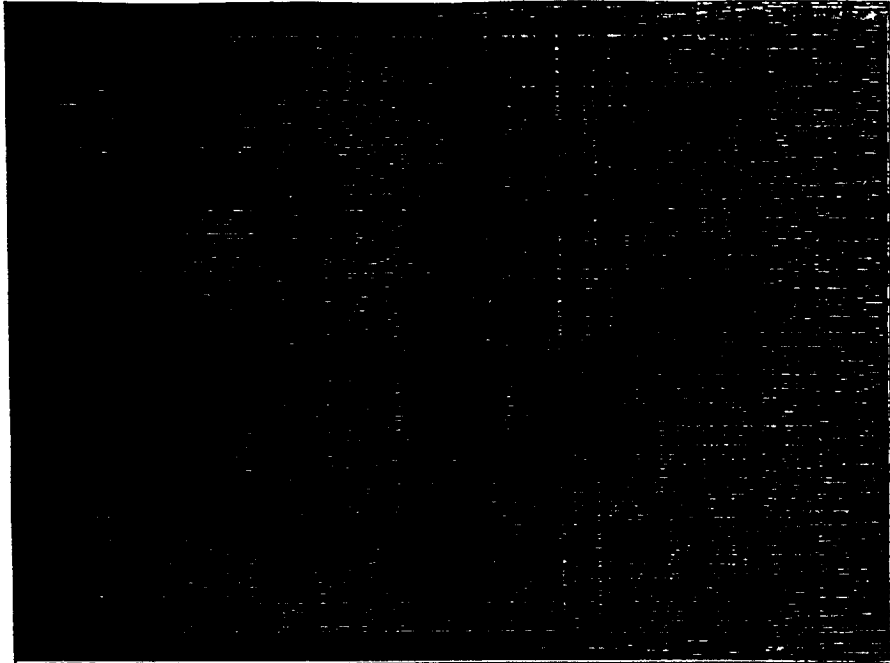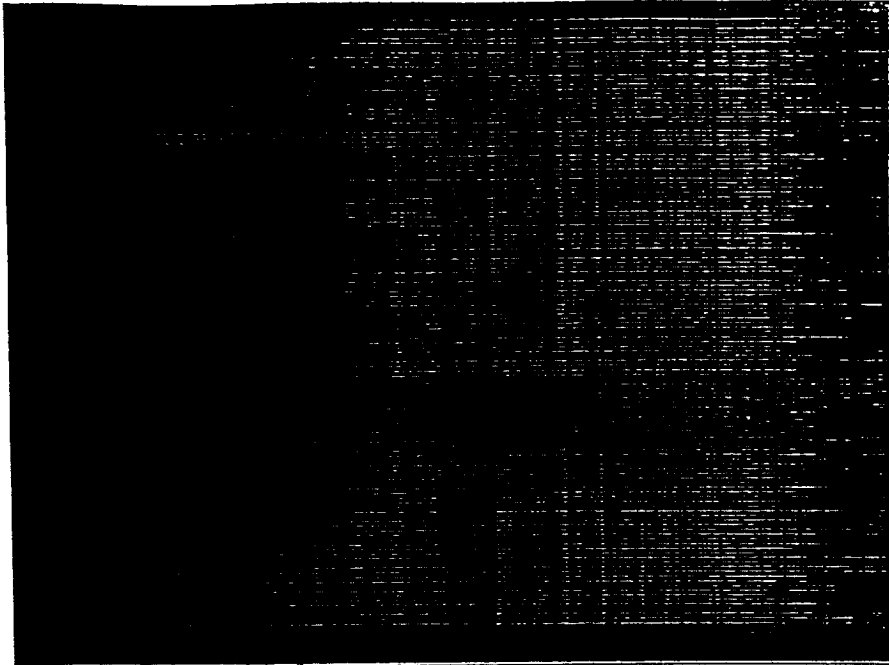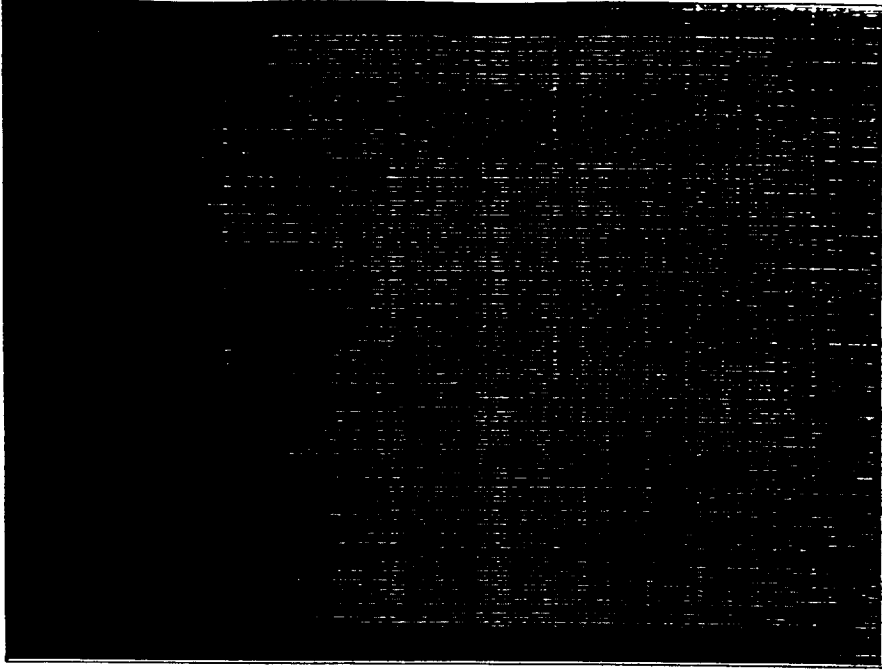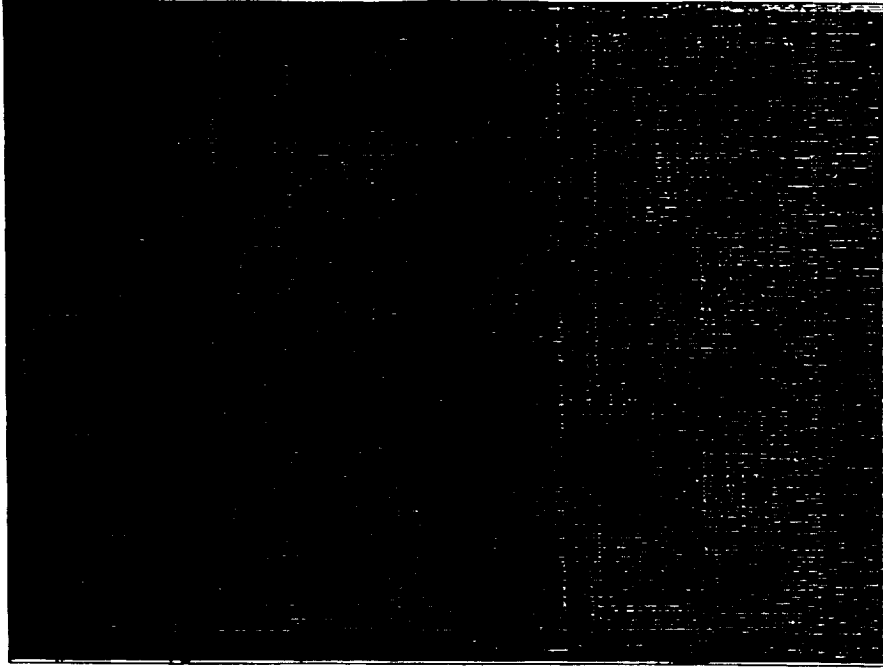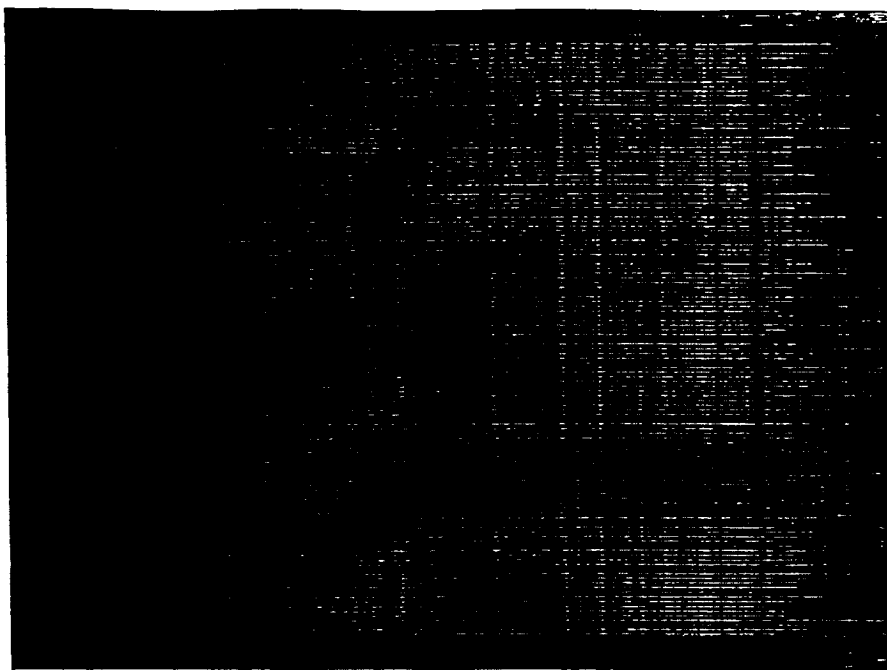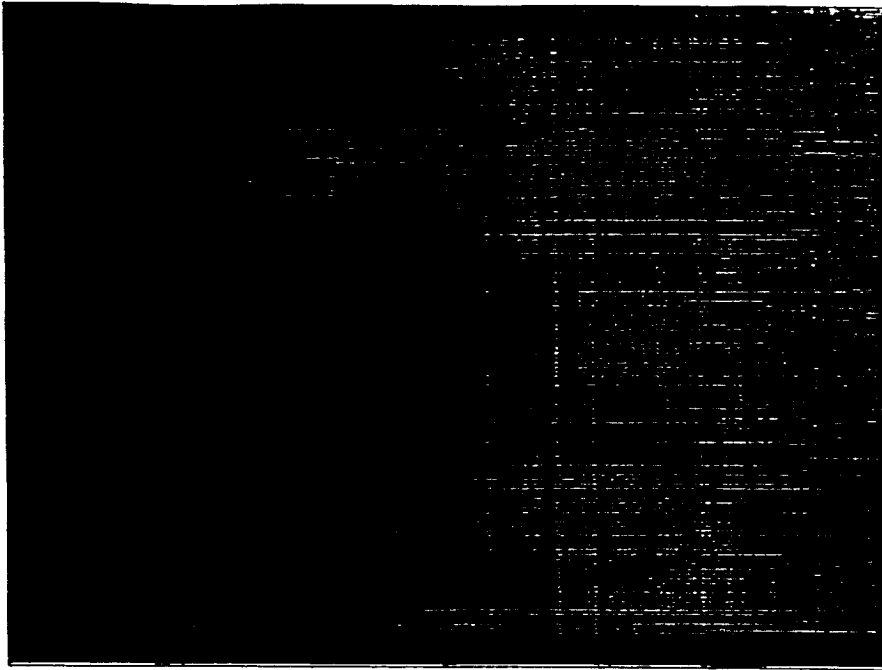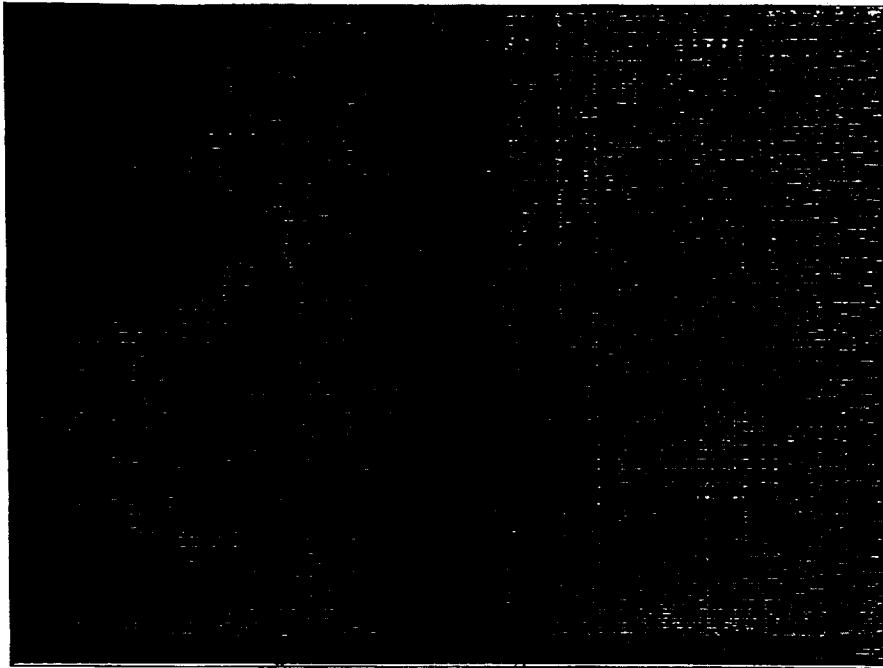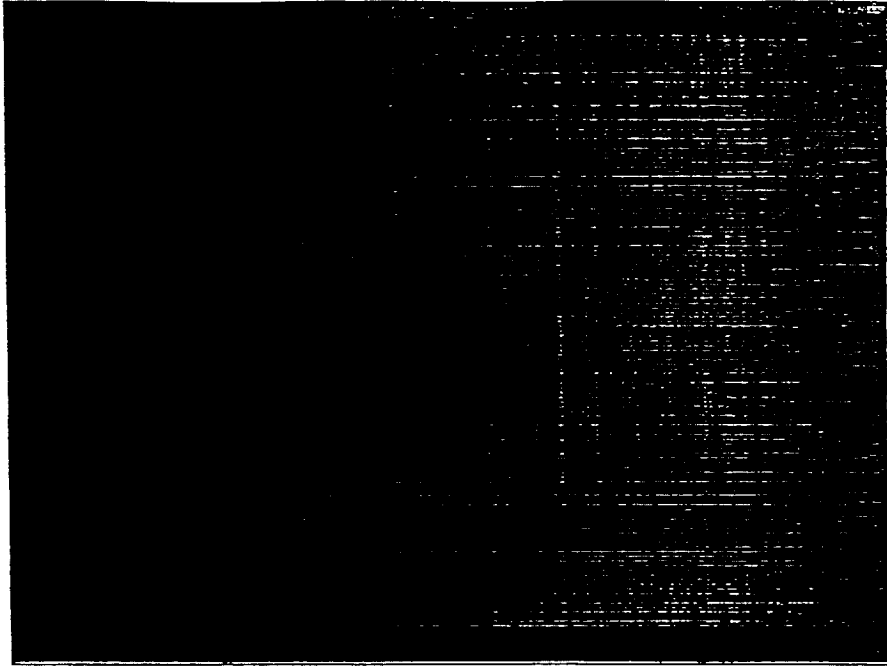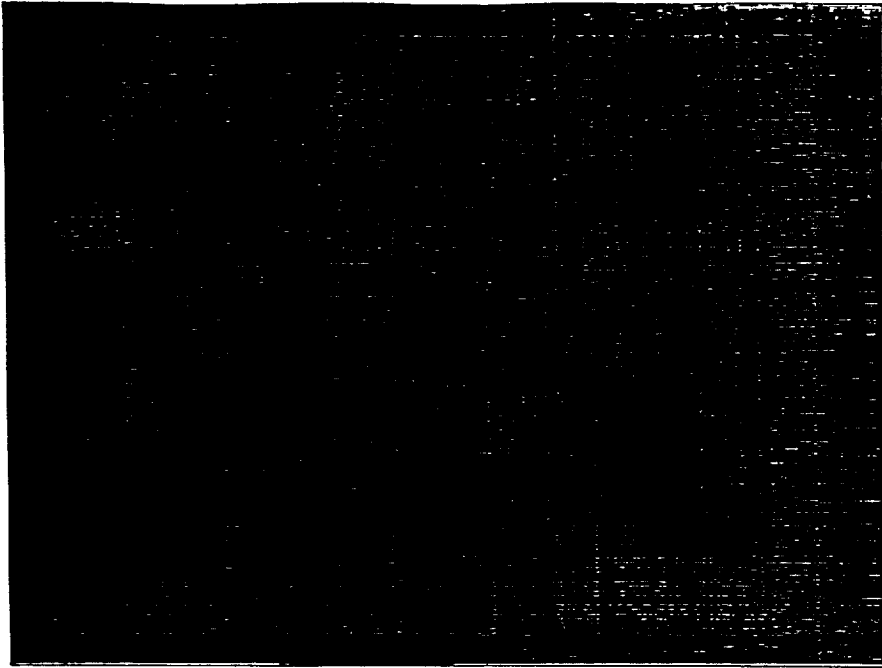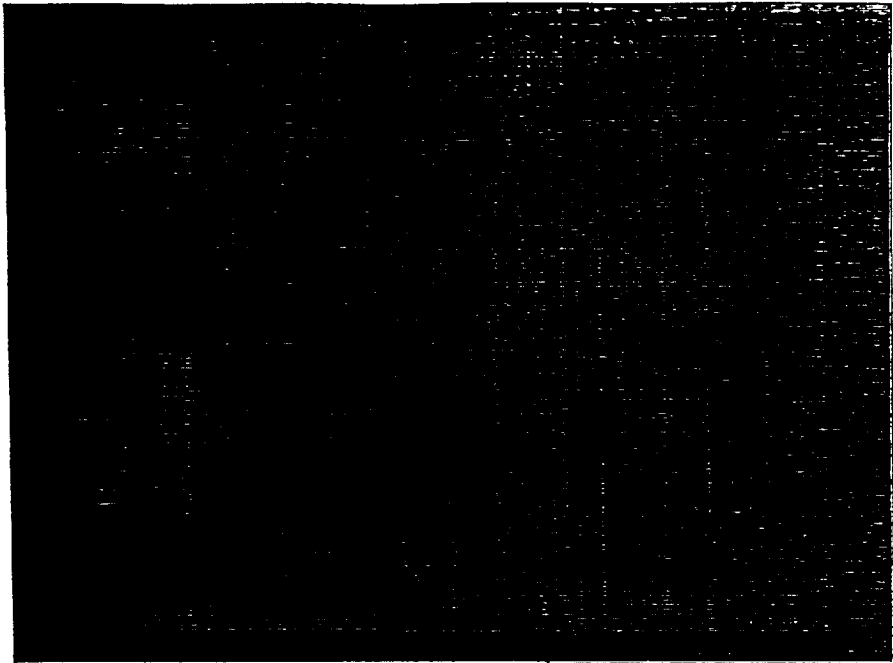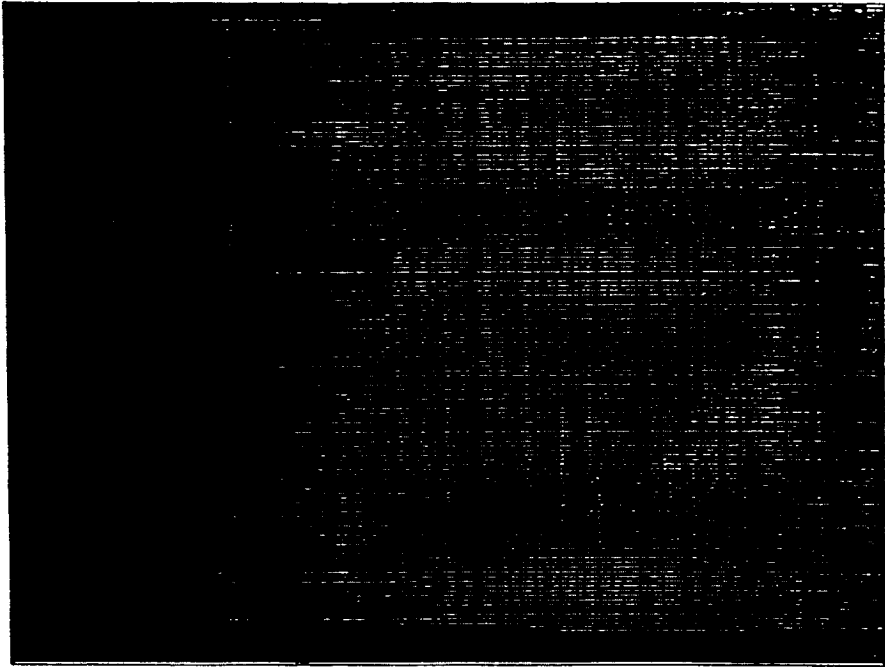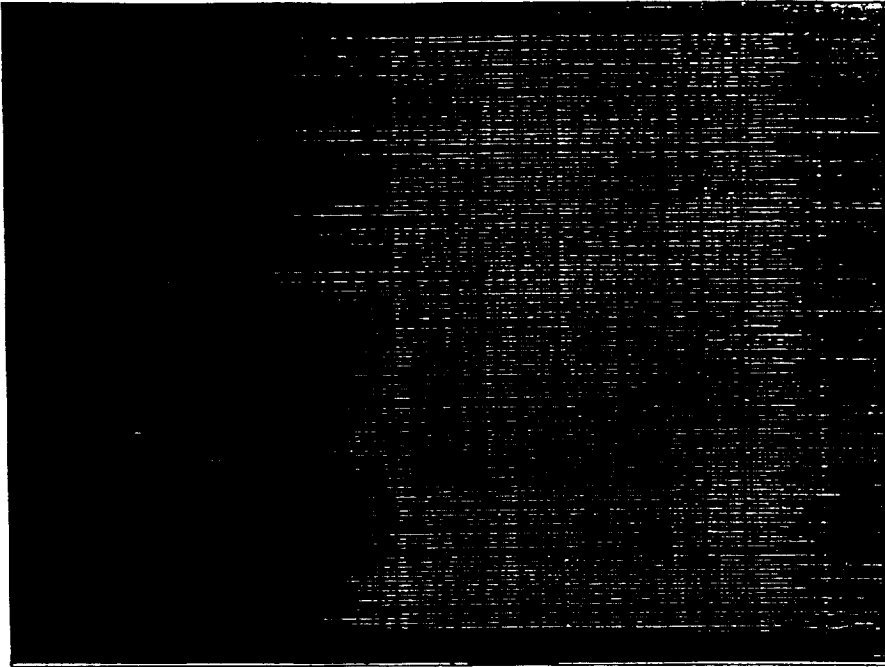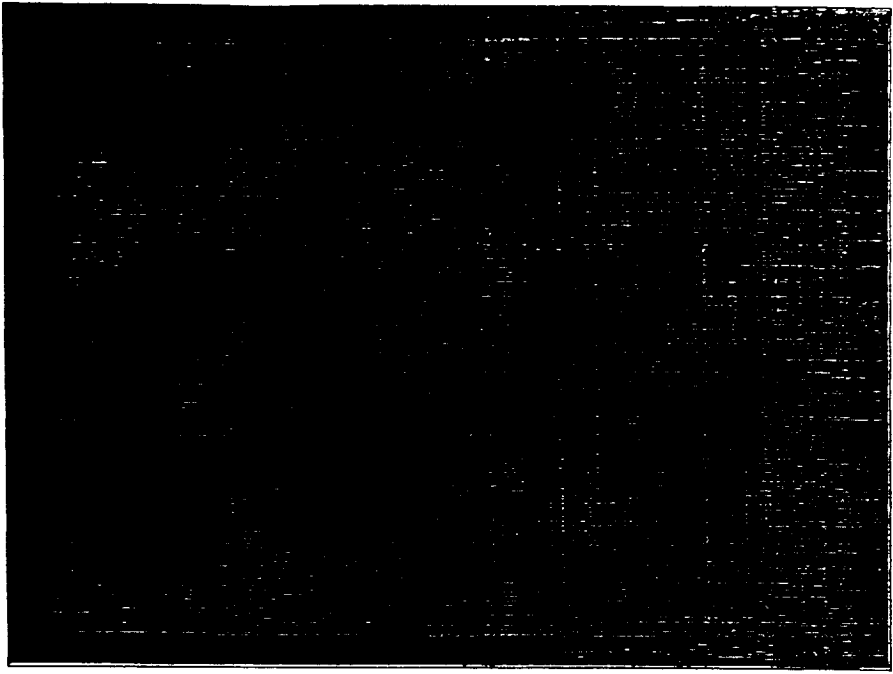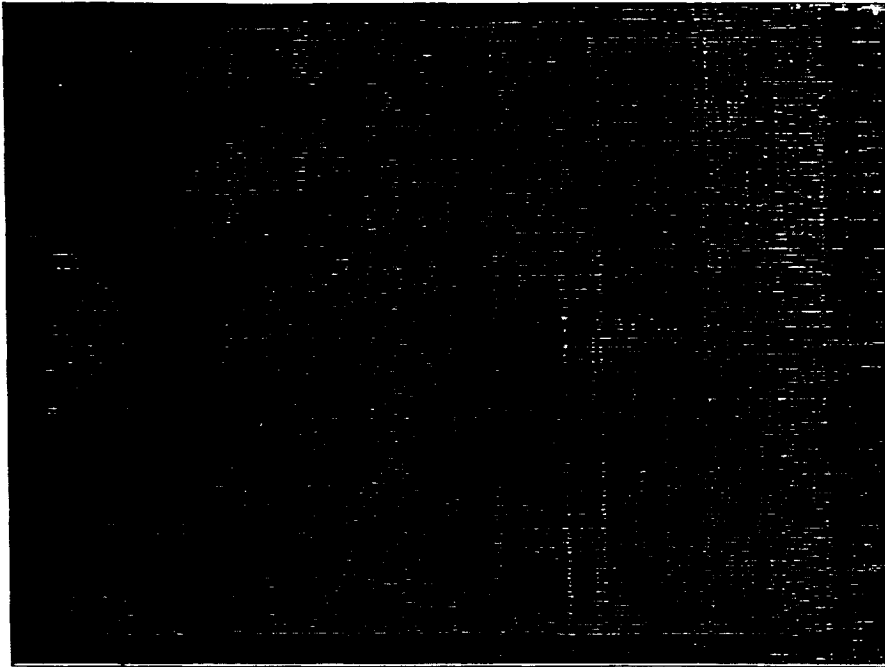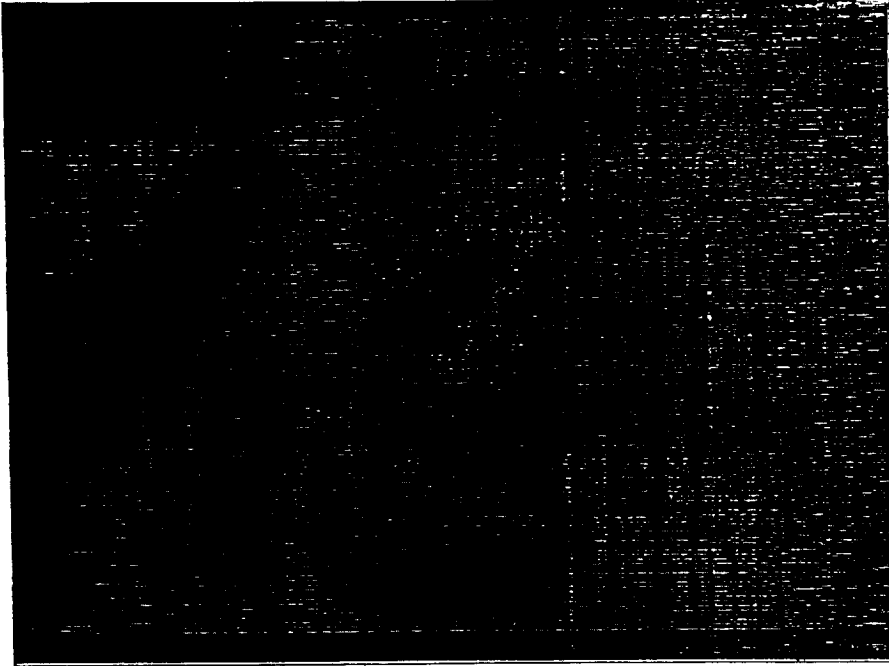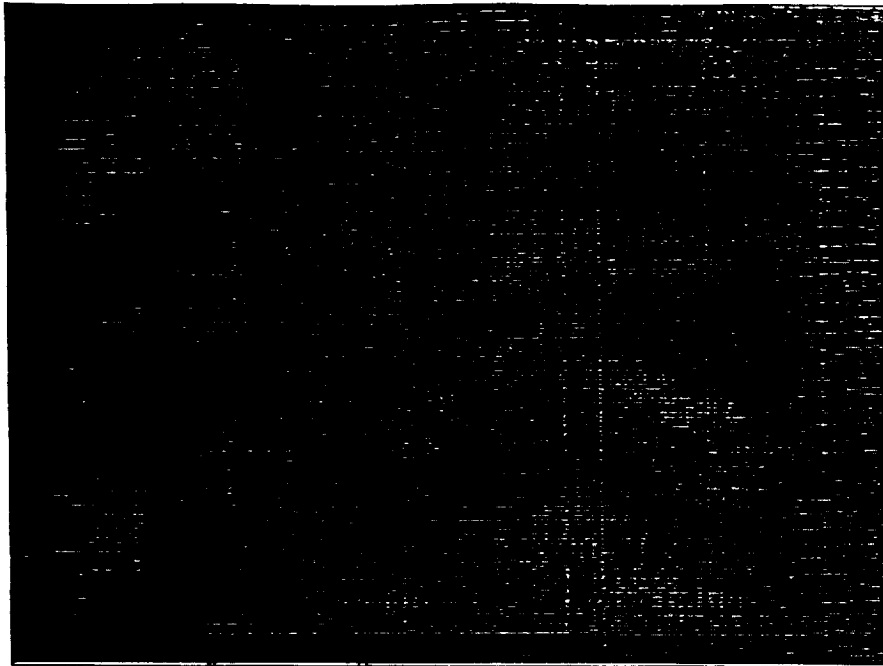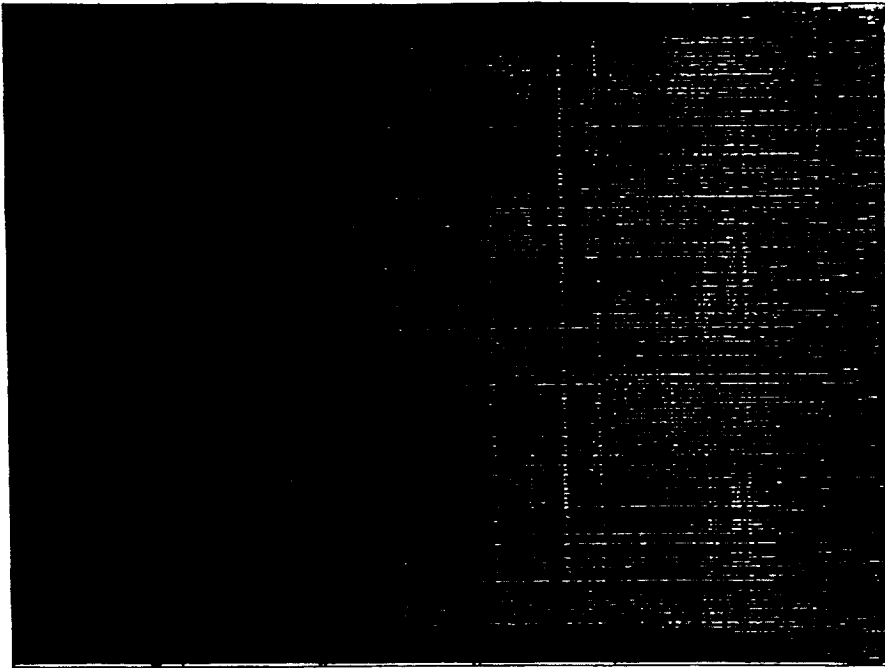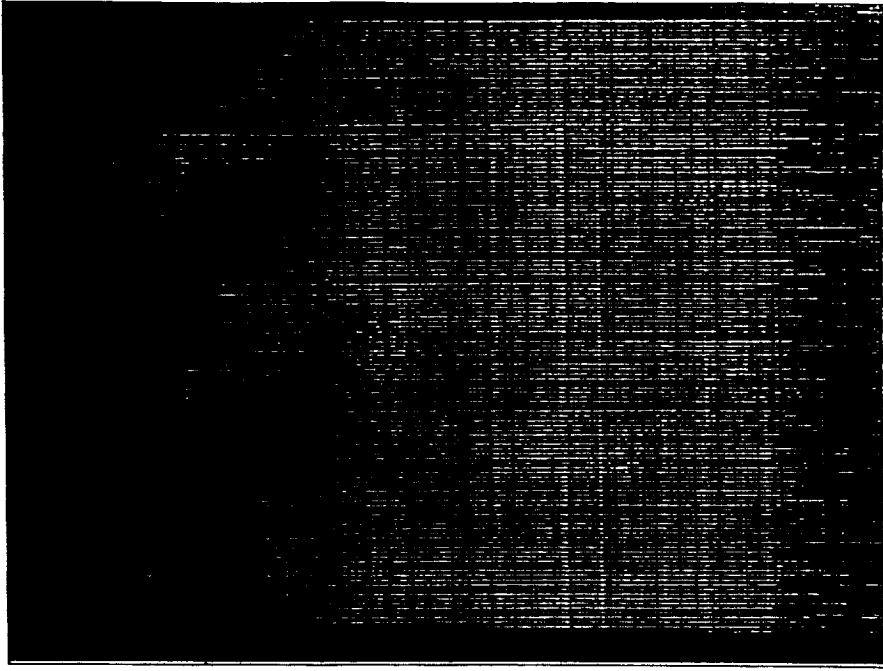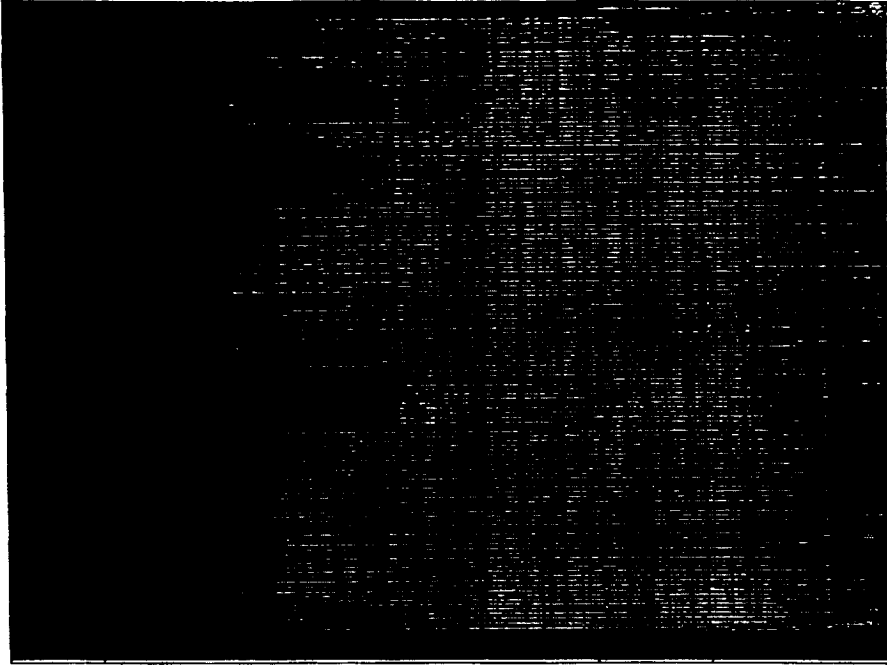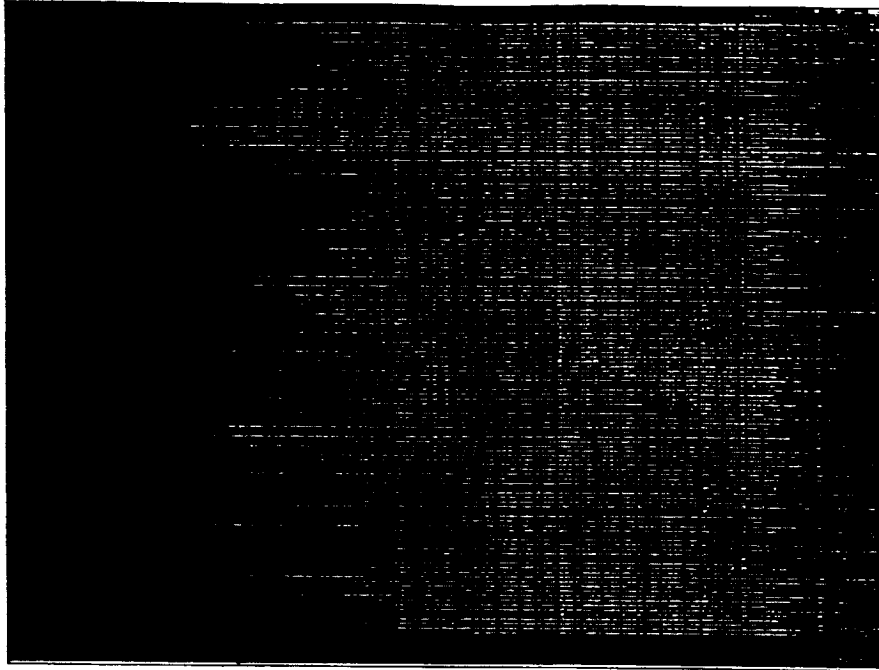
ATTACHMENT WILL NOT OPEN

Power Evangelism

James Plamondon

Technical Evangelist

Microsoft Developer Relations Group

In Our Previous Session

Evangelism Is War

Evangelism is WAR!

- ◆ **Mission**
  - ➢ Establish Microsoft's platforms as de facto standards

- ◆ **Enemies**
  - ➢ Other platform vendors

- ◆ **Battlefield**
  - ➢ ISV Mindshare

- ◆ **Progress**
  - ➢ Shipping ISV applications

The Role of ISVs

◇ **Pawns in the struggle**

◇ **Today's allies; tomorrow -- who knows?**

    ➤ We may move into their markets

    ➤ They may move into ours

◇ ***Valuable* pawns**

    ➤ We can't win without 'em

    ➤ Must take good care of them

◇ **Can't let 'em *feel* like pawns**

    ➤ Treat them with respect (as you use them)

Today's Topic

Getting ISVs to Do What You Want

- What do you want?
  - What does the ISV want?
  - What do you both want?
- Channels of Information
- Power, and how to use it

What Do You You Want?
Goals

◆ **Goal**

  ➢ Establish [platform] as the *de facto* standard in the industry

◆ **Different platforms; same goal**

◆ **Longer than a review period**

◆ **Not directly measurable**

◆ **So you also need objectives**

What Do You Want?
Platform Objectives

◇ **DRG does not create platforms**
  - ➤ Someone else in Microsoft does
  - ➤ What are their objectives?

◇ **Platform Business Plan**

◇ **Platform Product Manager**

◇ **They probably didn't think about ISVs or evangelism**

◇ **Turn their objectives into your objectives**

What Do You You Want?
SMART Objectives

- **Specific, Measurable, Actionable, Realistic, Timed**

- **Specific**
  - Win95 logo-compliant apps

- **Measurable**
  - 10 Win95 logo-compliant apps

- **Actionable**
  - 10 of 40 FirstWave apps (list)

What Do You You Want?
SMART *Objectives (cont'd)*

◈ Realistic

⋗ 10 (not all 40)

◈ Timed

⋗ by Spring Comdex '95

Resulting Objective:

10 of top 40 key apps showing
Win95 logo-compliant versions
at Spring Comdex '95

"Sold!" — Now, what?

◇ You need a complete infrastructure to turn a 'sale' into a shipping app

◇ Infrastructure:

➤ The resources that an ISV needs to deliver a shipping application that supports your platform.

◇ Building the infrastructure is part of evangelism

◇ Build the infrastructure *first*

Infrastructure

◆ **Decision-support materials**
  - ➤ Whitepapers
  - ➤ Demos
  - ➤ Testimonials
  - ➤ Analysts' reports

◆ **Training resources**
  - ➤ Documentation
  - ➤ Sample code
  - ➤ Books and articles
  - ➤ Courses

Infrastructure (cont'd)

- ◇ **Implementation resources**
  - ➢ **Consultants**
  - ➢ **Developers**
  - ➢ **Porting Lab**
  - ➢ **PSS**
  - ➢ **Tool support**
  - ➢ **Debuggers**
  - ➢ **Compliance Testing**

Infrastructure (cont'd)

- ◈ **Marketing Resources**
  - ➢ Events
  - ➢ Catalogs
  - ➢ Press Releases
  - ➢ Press Tours
  - ➢ CD Samplers
  - ➢ Joint Advertising

Mind Control

Mind Control

◇ To control mental output, to have to control mental input

◇ Take control of the channels by which developers receive information

◇ Then they can only think about the things you tell them

◇ Thus, you control mindshare

Channels of Information
One-on-One

◇ **Staple of Evangelism**
  ➤ **Effective, but expensive (time and $$)**

◇ **In-person**
  ➤ **Organize into a Jihad**

◇ **Videoconference**
  ➤ **All Microsoft sales offices are wired**

◇ **Telephone**

◇ **Conferences**

◇ **Tradeshows**

Channels of Information
One-to-Many

- ◆ Conference presentations
- ◆ Roadshows
- ◆ Tradeshow booths
- ◆ Developers Conferences

Channels of Information
Developer SIGs

◆ Special case of one-to-many meetings
  ➢ SIG leaders

◆ Go out with 'em afterwards

Channels of Information

Developer Conferences

◇ **Special case of one-to-many meetings**

◇ **Two kinds**

 ➢ **Controlled by platform vendor**

 ⟁ **Brainshare, PDC, LotuSphere**

 ➢ **Independent**

 ⟁ **SD, WinDev, MacHack**

◇ **Gather intelligence at enemy conferences; *be nice***

◇ **Subvert independent conferences**

 ➢ **Love them to death**

Channels of Information
Developer Magazines

- **Same as developer conferences**
- **Infiltrate and subvert**
- **Encourage new writers**
- **Add value to the independent magazines**

Channels of Information
On-line Forums

◆ Monitor the relevant Usenet groups at all times

◆ Write well

◆ Be exceedingly formal and polite

    ➢ It is very easy to give offense

    ➢ Always assume that you are wrong; ask others to explain it to you

    ➢ Developers are impressed by clear, precise, polite communication

    ➢ Don't sound like a prig

Channels of Information
Books

◇ **Very active book market**

◇ **Start the book *early***

◇ **Get a third-party to write it**

◇ **Consultants love this**

   ➤ Author == expert

◇ **Choose a credible author**

Channels of Information
Consultants

- ◆ Independent evangelists
- ◆ Must be on the bleeding edge
- ◆ "Patina of Objectivity"
- ◆ Contract houses
  - ➢ Sample code
  - ➢ Books, articles

Power

and

How to Use It

YOU have POWER!

- ◇ **Power**
  - ➢ The ability to GET THINGS DONE
- ◇ **Source of Power**
  - ➢ The ability to control the distribution of valuable resources
- ◇ **Exchange of Resources**

Exchange of Resources

◈ Trading favors

   ➢ "If you help me, I'll help you"

   ➢ ALWAYS return favors

   ➢ NEVER work with someone who has failed to return a favor

◈ Help people!

   ➢ Then they owe you a favor in return

◈ You have MANY resources

What Does the ISV Want?

◇ **Success**
  - Financial
  - Social/Status
  - Other

◇ **Corporate vs. Personal Success**
  - Stock price vs. salary
  - Long-term vs. short term
  - Good-for-product vs. good-for-resume

◇ **Identify the decision-maker's wants**
  - Then show how your platform fulfills those wants

Resources

◆ **Information**
  ➢ The ultimate resource

◆ **Specifications**

◆ **Betas**

◆ **SDKs**

◆ **Free Products**

◆ **Knowlege**

Resources
(continued)

- ◇ **Job Placement**
  - ▶ People looking for jobs
  - ▶ ISVs looking for candidates
- ◇ **Exposure**
  - ▶ Publications
  - ▶ Conferences
  - ▶ Seminars

Resources

(yes, you have still more resources)

◇ Cash

◇ Co-marketing

◇ Contracts

◇ Technical Support

◇ Sales Force

◇ Solution Providers

◇ Conferences

Resources
(an endless supply)

◆ **Exclusive events**

◆ **T-shirts**

◆ **Newsletter**

◆ **Developers Group**

◆ **Consultants Group**

◆ **Conferences**

◆ *Create your own resources!*

Summary

- ◆ You set the standard
- ◆ You have tremendous power
- ◆ Use it!
- ◆ Make things happen!
- ◆ Kick some ISV butt!
- ◆ Take no prisoners!
- ◆ *Windows, Windows, Windows!*