



Business uses of Peer to Peer (P2P) technologies

A Netmarkets Europe White Paper

Julian Bond julian@netmarketseurope.com,

© Copyright, Netmarkets Europe, January 2001

Business uses of Peer to Peer (P2P) technologies

Overview

Business to Business (B2B) trade is a potentially huge use of the Internet. The initial solutions to this via independent aggregating hubs are now being discredited as the early movers fail to generate revenue. The various alternatives of Industry Sponsored Markets (ISM), Private markets, Supply chain automation and others have their own problems particularly in the areas of content management, scalability and privacy. Almost all the thinking around this has been in terms of straight line value chains, or one to many aggregation dominated by big business. The smaller players and Small to Medium Enterprises (SME) have largely been ignored. It is now being realized that successful internet B2B systems are going to have to reflect the constantly shifting web of interconnections that characterizes business trade.

While this has been happening, the end of 2000 saw an explosion of interest in P2P systems driven by the huge take up of Napster and the large amounts of press this generated. While Napster (and it's clones) is all about finding and transferring files directly between clients, there are several ideas in this area which may well have application to B2B trade.

This document examines some of the current problems in B2B systems and explores ways in which a P2P approach might solve them. This is summarized in a brief proposal for a business P2P system that allows trade to happen directly between partners rather than via central hub.

Julian Bond julian@netmarketseurope.com,
Netmarkets Europe, 2001

Table of Contents

Overview	2
Table of Contents	3
Definitions	4
Centralization in B2B	4
The UI Problem	5
SMEs and Suppliers	5
Trust	6
Security.....	6
Catalogues	7
Revenues	7
Standards	8
A Modest Proposal	8
Flow diagram.....	10
Appendix - Potential Implementors.....	10
Appendix - Potential Implementors.....	11
Microsoft	11
Commerce One	11
Accounting Software	11
Upstarts.....	11
Appendix - Companies to keep an eye on	11
Appendix - Web URLs	12
About Netmarkets Europe	13

Definitions

There are two fundamental philosophies in internet systems. One is the trend towards centralization and aggregation typified by websites such as eBay, Yahoo and Covisint. This attempts to exploit network effects by drawing large numbers of people and large amounts of activity to a single system. But this suffers from decreasing returns as the very success that drives it also results in scalability and reliability problems that reduce the return for each participant. The other approach is a distributed, peer level, web of interconnections. This is usually built around a self organizing hierarchy. The classic examples of this are the fundamental systems that make the internet what it is. TCP/IP, DNS, email and Usenet. Arguably there's also a middle ground of hub & spoke models with the hubs linked. P2P is all about the second and third of these.

To me, "P2P" is nothing more than an acronym for systems which attempt to connect nodes together **directly** to pass information for **some** part of the process. It doesn't say anything about how they find each other, the nature of the information, whether the whole process is de-centralized or only part of it, what language the systems are written in, whether the software is given away or charged, whether it is open source and whether it is used for copyright bending.

Just because the most famous P2P implementation so far (Napster) is a system for sharing files doesn't mean that all P2P is about this. So, one of the problems for anybody trying to do P2P commercially, is going to be explaining how they're not doing just another Napster to the media and analysts. The specialists didn't figure out that P2P was about scalability, reliability and efficiency until late last year. The hipper analysts have now picked this up, but the me-too analysts - who depend on other analysts - will need a few more months to digest it all. Consequently, we can expect a lot of mis-reporting and confusion for some time to come.

Throughout this document I have used a new acronym, PB2B. This means Peered Business to Business Trade and implies systems that allow businesses to co-operate, transact and pass information by direct connections between them rather than via a central hub.

Centralization in B2B

Two to three years ago (late '98) eBay was making the transition from merely huge to massive. At the time everyone (media, analysts, commentators, practitioners) thought that auction and other dynamic pricing models were going to revolutionize business trade. This led to mistaken thinking about 'Land Grab' and hundreds of 'Me Too' auction sites. Eventually everyone understood that there would be no overriding winners and that auction was only appropriate for a very small subset of B2B trade. But the thinking behind "being as big as eBay" in a particular niche remained.

At about the same time, Dell and others were successfully selling direct and there was a lot of superficial thinking about dis-intermediation and driving efficiency into systems by cutting out the middlemen. Now distributors don't often add much value to a value chain, but it conveniently forgets that middlemen do provide credit and often bridge the gap between traders, usually by absorbing counter party risk or by applying domain expertise.

As thinking about netmarkets matured there was also a rash of comparisons with the financial markets. But these conveniently forgot that the financial exchanges have taken hundreds of years to develop and that most of the products where this style of trading was appropriate, were already served by established markets.

Today ('01), B2B thinking is focusing on e-Procurement and Big Business supply chains. But for reasons of scale and ease of implementation, this hasn't really progressed beyond MRO and indirect goods. Approaches to internet trading of direct manufacturing supplies are still in their infancy.

Despite the progress in other fields, there is still a common belief that there will be one winner in each industry where everyone will go to trade. This is the "Fat Butterfly" model where **All** the buyers and **All** the sellers end up in one place. This is the same 'Winner takes all' thinking but based on a hub and spoke model. This ignores the reality that in any industry you care to name there are now (say) 3 independent

markets, 3 rival consortia, 6 major players selling direct, 4 private markets trying to buy direct and thousands of large and SME suppliers trying to cope with the conflicting demands. We're now seeing a few initiatives to provide some punch through, pass through or information sharing between hubs to try and resolve this mess. The logical end point of consolidation in the B2B players would seem to be a single Exchange system for the whole world's trade, but this is clearly absurd. The actual end point is much more likely to be a constantly shifting, N Dimensional, "Indra's Net", web of point to point relationships. Just as it is in the real world of offline trading.

Now this sort of web of shifting and temporary interconnections is exactly what de-centralized P2P systems are good at.

So let's take a look at some other problems in implementing B2B trading systems and see if P2P would be a good approach to dealing with them.

The UI Problem

The web is an extremely successful system for publishing static and even quasi dynamic information. The key word here is "publishing"; One way posting of information by One central site for consumption by the Many readers. There are a few problems with HTML's lack of control over exact representation, but it has the huge benefit of being largely browser neutral. The information is readable and accessible by systems with a very low cost of entry.

But trade is inevitably going to require a considerably more complicated user interface. The users will need to search, scroll through records, enter data and generally collaborate between the buyer and seller in the process leading to and from a transaction. Unfortunately, HTML CGI is an appalling interface for real applications. To take a trivial example, compare something like Hotmail with any half way decent email reader. The local reader has a scrolling search mechanism and a real editor with real time spell checkers and such like. Viewed in this light the whole idea of doing mainstream work in HTML based ASPs is laughable and attempting to complete business transactions using just HTML equally so.

The Java Applet alternative to this is not much better with large downloads and slow load times. Local Java implementations are also pretty unpleasant to use compared with native code and systems that use the local OS APIs. Then there are the problems with the different implementations of Java leading to some nasty upgrade problems. This can all be dealt with in a server implementation, but looks insurmountable as a basis for client side applications.

Having said that, the one good thing about HTML and most Java is the zero installation. If a PB2B system is built as a local application (on MS Windows), then it is likely to involve 5-10 Mb of initial install file. For a once off installation this is probably acceptable. We also have to deal with the upgrade and enhancement process but I think it's reasonable to say that Real.com and IE has shown that providing the application is properly architected, we can deal with online and automatic upgrades, extensions and point releases.

So it would seem that a downloadable native application is probably the only way to provide a system for B2B trade that is truly useable.

SMEs and Suppliers

The current state of the capital markets mean that the majority of the B2B vendors are chasing the global 10,000 and moving away from the independent markets. The most obvious use for the netmarket technology that they have built is for the procurement and buying processes within these large corporations. Of course this is a double sided coin because to make this work, they also have to sign up the suppliers. Many of these suppliers are SMEs and in every country in the west, these SMEs represent at least 50% of the workforce and trade. The larger suppliers have already put a lot of effort into building their own online selling systems that are tightly coupled to their own back office systems.

Clearly at some stage in the future, all these SMEs are going to be involved in online trade. At the moment the most common proposal for them consists of providing instant web store front technology with minimal integration into their back office. When dealing with big business, these same SMEs are expected and

being asked to provide their catalogues to all the major buyers and take part in a process that also doesn't integrate into their own systems. There have been few attempts to build real online systems for the SMEs because customer acquisition and support is seen as too expensive. And let's not forget that an SMEs "Back Office System" is unlikely to be more than a copy of Quicken or Sage operated by the owner's significant other or even on paper.

Suppliers have also typically spent a large amount of money (relatively) on building a reputation and brand. The big buyer market systems expect them to submerge this into just another line item.

And yet, we're finally reaching the stage where any real company is going to have some form of always on internet access with reasonable bandwidth. This opens up the possibility of them running their own server on their own premises. Which in turn should make it easier to link this direct to their back office system without introducing awkward security problems.

A small footprint PB2B system aimed specifically at SMEs has got a real chance of solving some of these issues.

Trust

One of the supposed benefits of B2B trade systems was transparency of pricing. In a public market, it may not be possible to see exactly who is paying what, but it was hoped that participants would be able to see what prices are being paid in the market. This is part of the early economist's theories about perfect information and perfect markets. But in reality, in all but the most liquid markets, this makes suppliers very uncomfortable because it benefits buyers more than sellers.

There are many economies round the world where organizations are bad at paying, taking anywhere from 60 to 360 days. Consequently all trade is done on account. Which in turn means that trade is impossible without approval from the credit department. So, the whole idea of spot trading for cash in a neutral exchange becomes impossible without some third party providing the credit (Remember dis-intermediation?).

It's also common to have a guide price list but to have numerous exceptions and discounts for classes of customers, temporary special offers and so on.

To put this another way, trade only happens where there is a pre-existing relationship and much of the pain comes from building these relationships initially. Further, channel agreements mean that there are frequently overt rules about who can trade with who and at what price. But covertly, these are equally frequently broken.

All this suggests that our proposed PB2B system must support something like buddy lists with differential availability and pricing for the different lists and members. All this is extremely hard to implement and maintain in a public market, but a little easier when it's in a local system. ICQ provides a good example of how some of this might work.

Michael Fertik (TrueXchange) has summed up some of this as "In Business, dis-information is often more important than information".

Security

The next part of the centralized B2B problem is the security and privacy aspects. Even with SSL, many companies feel very uncomfortable about their transactions being held on a central system and nobody is really addressing this yet. PKI and similar systems are now pretty well understood and provide Privacy, Non-Repudiation and Proof of Integrity. But trying to implement these on a central system for all participants along with privacy from the site's owners is non-trivial to say the least. Actually implementing any of the PK systems on a client site is equally awkward.

By routing the transactions direct (P2P) we have an opportunity to fully encrypt and sign the details using a built in PK system using public domain code libraries that have been properly peer reviewed by encryption experts. Since the data never leaves the control of the two parties as plain text, we can ensure complete

integrity. The organizations can decide unilaterally how much or how little to reveal both to each other and to the world at large.

Catalogues

One of the bigger problems in B2B is the Catalogue problem, and it's big! When we say Catalogue, we mean the Product Description, Pricing and Stock Availability from suppliers and the Product Description, Pricing and Delivery Requirements from buyers. The traditional B2B approach to this is to try to aggregate the catalogue from multiple suppliers in the hub and then to index and categorize it so it can be searched and matched. But:-

- Nobody can agree on how to describe a general product so every hub creates it's own standards
- Nobody can agree how to categorize products so every hub also creates it's own standards. Global category standards are being developed but several of them seem to have been produced by people who didn't even do DBMS 101. Then there are several competing standards initiatives with no clear winners yet.
- SMEs find themselves being told to provide their catalogues in new forms (often using XML) and having to pay themselves for the conversions. If there is more than one hub in an industry they will have to do this several times.
- There's a huge back log at most hubs with suppliers being told that they are (say) #250 in the upload queue and that they can only have 4 updates a year. (A real world example from a major hub)
- The source data is frequently on paper, badly formatted spreadsheets or buried in a back office system.
- Those back office systems range from SAP to paper and everything in between.

So taking multiple copies of the source data to aggregate it into multiple hubs looks nonsensical. How much better would it be if the data was left where it was and accessed directly as needed. But we would still like to provide a search across all the data to locate products and suppliers. The traditional way to do this is to use directories such as Yellow Pages, or now web portals or search engines to locate suppliers who might carry the product. Then we would do some initial searches on each website and draw up a short list of possible suppliers.

Consider the nature of the problem. All the databases are large. There are maybe 10^6 manufacturers and suppliers and 10^8 product SKUs (to within a few orders of magnitude). And we're talking about needing a distributed data search across all this. It hardly needs to be stated that this is non-trivial!

A criticism of an automated solution to this is that when looking for web pages, we are not too upset if we miss a few because our search terms were badly considered. But in a business transaction we cannot afford to miss a product supplier for the same reason. Well this is not really any worse than real life. Short lists are frequently arrived at for emotional reasons in the absence of perfect information.

One possible solution to this is to do the search in two phases. The main search system could work on a superficial description of the products using full text searching. This is remarkably like the search in Napster. Then a deeper search could be done direct to the supplier's sites on the short list. This is one area where there is a great deal of P2P activity. Gnutella, Freenet, Infrasearch and Kazaa are all trying to solve this problem and advance the state of the art.

At the same time, the IBM/Microsoft/Ariba UDDI initiative is attempting to come up with a distributed index for internet site services allied with the SOAP standards for accessing those services. If this was broadened to include products, it might also serve as the basis of a solution to the distributed catalogue problem.

Revenues

Perhaps the biggest problem I can see with a completely de-centralized system is that the lack of a central point is going to make it pretty much impossible to extract revenue from the activity in the system.

Now for some of the players in this game, revenue is not really a requirement. For others 'Volume is it's own reward' and they'll worry about the revenue later. But for the rest of us, having the system generate real revenue provides a big incentive for implementing something.

So trying to get revenue from the activity looks like a non starter. It's hard to imagine anyone being happy with paying a toll for activity. Not least because one of the advantages of a peer system is the inherent privacy. If the activity is hidden from the world at large, how are you going to know what happened and "tax" it.

My guess is that the simplest way to generate revenue is from software sales and upgrades. But this may work against our desire to get rapid takeup. The best solution to this I can see is the Netscape or Real model; give away a fully functional but limited Client, but charge a subscription or upgrade fee for the Pro system, Server and plug-ins. There is a downside though. If we decide that open protocols are important (I believe this is true) then the standard setter is going to want to actively encourage competition for the software. Something that the software companies may not be entirely happy with.

The advertising model might be attractive, but remember Pointcast? The same competition for clients is going to immediately result in a competitive system with the adverts removed.

Standards

The chances of creating another eBay or Napster style success are non zero but highly unlikely. If a particular implementation is hugely successful, we will get "De Facto" standards as a by-product. Trying to force this is not going to work as creating De Facto standards on demand is something even Microsoft can't guarantee. So we are probably better off ignoring it. Still, if we're going to get widespread adoption, we need a number of things: a viral element; competing implementations; sufficient standards to support this.

I'm unsure of where the standards are going to come from. I have an inherent distrust of Industry standards bodies because there are generally too many hidden agendas. So they either move too slowly or the standard becomes too complex. Good standards (like the internet infrastructure) are usually small, tight and succinct and are developed by a benevolent dictatorship "because it's neat thing to do".

Luckily the Internet has a long history of exactly this sort of standards activity. Where it works best is where the standards are relatively low level but allow complicated implementations on top of them. Our PB2B system is going to need some transport, data description and remote invocation standards but as long as we keep this simple and extendible, it feels achievable.

A Modest Proposal

So far I've presented a series of B2B problems and the beginnings of some P2P solutions to them. It appears that there is some potential to create a generic P2P trading system that is targeted at the middle ground of suppliers and buyers and which includes a generalized product search across the whole system.

Picture a system where each end user builds and controls their own local database. This gets away from the need to copy the data to a hub. Because the local data is locally controlled it can be linked directly to the local back office and stock control system without introducing too many security concerns.

This is then published to a global index system (centralized or de-centralized) that allows the global database to be searched at a first general level, returning a list of sites with products that might fit the search. A peer search then queries each site directly to get a fully accurate list and then to get full product information. This two stage approach can simplify the global index and reduce it to 5 or 10 attributes rather than the 10-100 attributes required by typical products.

To take a real world example. Say I'm looking for 100 metres of cold drawn welded steel tube in 2cm diameter and 16swg wall thickness for delivery in N.A. for a particular project. Steel tube returns the 100 suppliers in North America from the global index. I kick off the second search and it's reduced to 10 based on the quantity and three based on the metric sizes and type as well. Now I have direct contact with the three suppliers who can meet my need. From these, I view the full product description sheets and dismiss one because they're out of stock.

I can now start a negotiation direct with my two suppliers and either manually (or with automation) play them off against each other.

Let's imagine an implementation. We're going to need a number of pieces that might be provided separately or bundled.

- 1) A combined Client/Server for small business
- 2) An Index server that can be distributed
- 3) A Server for larger business that would straddle their firewall
- 4) A Client for larger business that talks to their local server

#1 Looks quite Napster like in that it is both the search/transaction tool but also the public face of the system. It probably has an MS Windows interface with alternate implementations for X and Mac. It has plug-ins for the common accounting/stock control systems like Quicken, Sage, JD Edwards and Great Plains. It might be written in Java or C++.

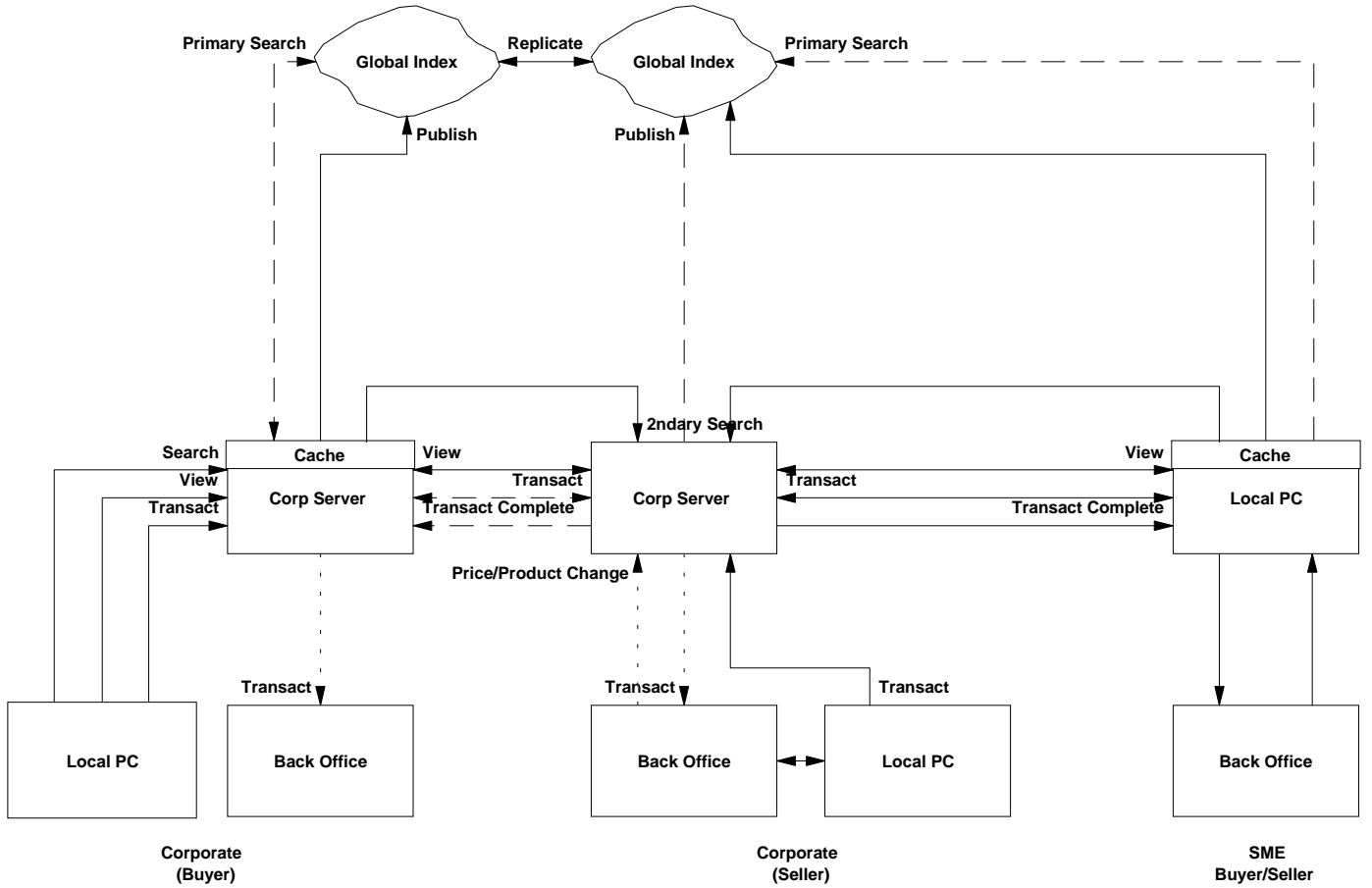
In #2, the Index system will have to be distributed for scalability but could range from the sort of hierarchical approach with local caches exemplified by DNS, through the anarchic but controlled SuperNode approach of Kazaa, to the completely distributed approach of Gnutella/Freenet/Infrasearch. It's important though that there should be source code available and that anyone who wants should be able to implement a server and join the search network. This allows the growth to be market controlled and free from dominance by any one player.

The split Client and Server of 3),4) are really just expansions of 1) but recognise that a corporate will want control over who and what publishes information and has access to the local back office. By straddling the firewall it avoids having to expose the back office systems directly and it also allows for multiple people in that organization being able to use the system with appropriate permissions. The client UI could be HTML or heavyweight Java as an alternative to Windows as it's likely to have a LAN link to the local server.

This approach also provides plenty of opportunity for future expansion as we can use the basic framework to include features like online collaboration. Equally, the product view processes could be extended to include transfer of engineering drawings, documents and other forms of design collaboration.

By including PKI encryption in the links we can ensure privacy, non repudiation and integrity. By splitting the client and server we can satisfy business security. Finally we give the data owners complete control about what data they expose and to who.

Flow diagram



Appendix - Potential Implementors

This appendix lists a short collection of people who might produce a PB2B system as described above.

Microsoft

Microsoft is building an impressive collection of capabilities. They are now involved in the UDDI, Biztalk and SOAP standards initiatives, a mid range accounting package in Great Plains, a small business portal in bCentral along with their Commerce Server and Internet Information Server. Finally they have announced the .NET plans. Combining these with their control of MS Windows and Internet Explorer positions them very well to capture B2B and B2C activity for mid range and small organizations. They have made no specific announcements about P2P (that I have seen) but some of this and particularly UDDI could begin to look very like the sort of system I am proposing.

Commerce One

C-1s systems so far have been largely aimed at eProcurement in big business. Although this is quite centralized in approach, two areas are interesting. Their server side software is MS Windows based and installed on every C-1 customer's site. Secondly, they are attempting to aggregate all their customers into a global indexing system. This positions them well to produce a BP2P system. It remains to be seen whether they decide to go down this route.

Accounting Software

The mid range accounting packages, JD Edwards, Great Plains (See Microsoft above), Quicken and Sage have a stranglehold over the back office systems of SMEs. Sage has already started to try and leverage this by creating an Internet portal for SMEs with traditional ISP facilities for their customers. As the source data for B2B system is already held in these systems, they are in a good position to set some standards for exposing this data and for facilitating the automatic flow of transaction information in and out of it. But then this will take them out of their traditional business and so may be ignored. But any PB2B system is going to have to work with these companies to ensure good integration with them.

Upstarts

As always there is a chance of a completely new software company appearing that takes the market by storm. The problem as ever is to generate enough momentum quickly enough to make a difference. The various P2P web sites are now listing 70-100 companies in this area although very few of them are looking at specifically B2B systems.

Appendix - Companies to keep an eye on

- Aimster
- Consilient
- Endeavour
- Groove Networks
- Kalepa Networks
- Kazaa
- Killdara
- Radio Userland

Appendix - Web URLs

<http://www.peertal.com>

<http://www.p2p.freeservers.com/>

<http://www.peerprofits.com/>

<http://www.gldialtone.com/>

<http://p2pmania.dyndns.org/>

<http://www.oreillynet.com/p2p/>

<http://www.b2business.net/infrastructure/Peer-to-Peer/>

<http://www.cpfr.org/>

<http://www.vics.org/>

<http://www.kazaa.com>

<http://www.zeropaid.com>

<http://www.gonesilent.com>

<http://www.metiom.com>

<http://www.groove.net>

<http://radio.userland.com/>

<http://magi.endeavors.org>

<http://www.egroups.com/group/decentralization>

<http://www.egroups.com/group/p2p-discuss>

<http://www.egroups.com/group/peerprofits>

About Netmarkets Europe

Netmarkets Europe is the trading community for the B2B e-commerce industry in Europe.

We help our members maximise their ROI from complex e-business projects.

We do this by providing online and offline services that enhance their ability to Learn, Collaborate, Innovate and Trade.

Our services include a virtual "Knowledge Management" infrastructure, coordinating a pan-Euro trading exchange, providing private advisory services, Brokering deals between members, Setting and monitoring standards for this nascent industry.

- Specifically, we configure the following products into integrated programmes on relevant 'hot topics':
- Electronically-enabled face-to-face Workshops
- Networking events
- Email Newsletters
- Website portal focused on B2B
- Virtual discussion forums
- Online searchable Directory of service providers integrated with an RFP (Request for Proposal) engine.

We have been operational since May 2000 and have meaningful partnerships with, among others, NetMarketMakers (US), Newmediary (US), First Tuesday (global), Ernst & Young (UK). More will announced in the next month.

For more information or for sponsorship opportunities please contact Simon Torrance on simon@netmarketseurope.com (www.netmarketseurope.com)