

Ivip (Internet Vastly Improved Plumbing) Conceptual Summary and Analysis

Robin Whittle 2009-04-15 (See Changelog at end.)

Document history and structure

This is an update to the 8 page Conceptual Summary and Analysis document I prepared for the Routing Research Group in February 2008. It is now longer than 8 pages. For the context of this discussion of a new routing and addressing architecture for the Internet, to solve the routing scalability problem, please see: <http://www.firstpr.com.au/ip/ivip/> (which links to all Ivip Internet Drafts) and the IRTF Routing Research Group page and mailing list archives: <http://trac.tools.ietf.org/group/irtf/trac/wiki/RoutingResearchGroup>.

The core of the document (“Conceptual Summary” to “Analysis wrt RRG Design Goals”) remains that of the February 2008 version, with only minor improvements. Following this, is a new section on the use of “Forwarding”, as an alternative to “encapsulation”.

“Modified Header Forwarding” (MHF) instead of encapsulation in the long term

Ivip is part of the “core-edge-separation” class of scalable routing solutions. Such solutions usually involve “ITRs” (Ingress Tunnel Routers) and “ETRs” (Egress Tunnel Routers) with a method for tunneling traffic packets from any ITR to any ETR.

Ivip was initially conceived of as a “map-encap” (map-encaps) scheme, as are the other major scalable routing system proposals: LISP, APT and TRRP. It is probably easiest to explain Ivip initially in these terms. However, in August 2008, I developed two alternatives to encapsulation:

ETR Address Forwarding (EAF) - for IPv4
Prefix Label Forwarding (PLF) - for IPv6

These involve modifying the structure of the existing IPv4 or IPv6 headers – and upgrading most or all of the DFZ routers to handle this format. This may be relatively easy to achieve, since it may involve only firmware upgrades a limited number of types of routers, primarily or wholly from Cisco and Juniper. No changes are required to BGP in order to support “Forwarding” across the DFZ.

For IPv4 and independently for IPv6, there may be a period of encapsulation, in which Ivip runs wholly or primarily as a map-encap system. However, map-encap is not the long-term goal of Ivip. Ideally, there would be no encapsulation and Ivip would be implemented for both IPv4 and IPv6 using MHF alone. While these techniques are available to other scalable routing solutions, to date, Ivip is the only one which incorporates them.

MHF eliminates considerable protocol and functional complexity in ITRs and ETRs, due to the elimination of PMTUD (Path MTU Discovery) problems which are inherent in map-encap. MHF also eliminates the overhead of the map-encaps encapsulation header, which is 20 and 40 bytes for IPv4 and IPv6 respectively. MHF may also have some other applications. For instance, PLF for IPv6 can be used within each ISP for a private forwarding system, similar to that used globally in the DFZ, in both cases with 2^{19} destinations.

There are two approaches to this, for both IPv4 and IPv6. Firstly, Ivip (for instance for IPv4) could be introduced incrementally, with map-encaps, without the need to upgrade DFZ routers. However, all ITRs and ETRs would be capable of using MHF so that once the DFZ routers are all upgraded, the system would stop using encapsulation and switch over to “Forwarding”. The other approach is to upgrade the DFZ routers in time for the introduction of Ivip, and thereby deploy it using MHF exclusively. This would be technically much simpler, since it avoids the need for developing some complex protocols and

for implementing them in ITRs and ETRs. Given the lesser urgency of solving the IPv6 routing scaling problem, and the lower number of DFZ routers (perhaps more modern routers too) which handle IPv6, it seems highly likely that IPv6 Ipvip could be introduced initially with MHF only.

Conceptual Summary

Ivip is proposed as a complete solution to the routing scalability problem, for both IPv4 and IPv6. It is also intended to assist with the IPv4 address depletion problem by enabling greater utilization of address space, and to provide a new form of mobility for both IPv4 and IPv6.

The documentation of Ivip is currently in a state of flux. I intend to replace the long *ivip-arch-00* Internet Draft (I-D) by a shorter high-level architectural outline, which will cite this document and others, and form the anchor point for other Ivip I-Ds. Currently, this I-D is the best source of information on most of the proposal, other than “Forwarding”, the PMTUD and Fragmentation aspects, which are documented in the *pmtud-frag* directory of the Ivip home page: <http://www.firstpr.com.au/ip/ivip/> and the fast push database distribution system, which is described in the I-D *ivip-db-fast-push*. The Ivip documentation and some of the details of the proposal will be improved as soon as possible with new I-Ds concentrating on particular subjects, including ITRs and Query Servers; ETRs and filtering; Mobility; and Deployment. The Ivip home page links to a 25 question comparison of the various map-encap proposals and to RRG messages regarding Mobility and business cases.

The following terse architectural overview is intended for RRG members who have kept up with mailing list discussions in recent months. *ivip-db-fast-push* also has some overview material which may be helpful to newcomers. For brevity, IPv4 is assumed in the following description, except where noted.

A New Type of Address Space

Ivip provides a new type of address space which is optimised for the needs of end-users large and small - meaning organisations and individuals who require Internet connectivity for any purpose other than resale. This is referred to as Scalable PI - SPI - space. SPI space will generally not be used by ISPs, who will continue to use their own assigned space with conventional BGP advertisements.

In order for a routing scaling solution to be effective, it must be *voluntarily* adopted by the great majority of end-user networks who desire multihoming and/or portability of their address space between ISPs. Otherwise, those which don't adopt it will continue to use the existing BGP approach to multihoming and portability, using conventional PI (Provider Independent) prefixes. It is agreed that these PI prefixes is the primary source of growth in the number of BGP routes which a successful scalable routing solution must strictly limit. Since SPI space - which avoids this growth in BGP routes - cannot be imposed, a successful solution must make it attractive to end-user networks of all sizes and types. If it was not attractive to “larger” end-user networks, then many smaller ones would not adopt it, since their administrators aspire to be running them as “larger” networks in the future.

SPI space will be created from address space which is currently managed conventionally by BGP. Each prefix of the global public unicast address space which is managed by Ivip to provide SPI space will still be advertised in BGP. A single such BGP-advertised prefix is known as a Mapped Address Block (MAB). There are a variety of business models under which this could occur, but in principle, each MAB should provide SPI space for multiple end-user networks - for potentially thousands of large and small end-user networks. In this way, instead of each end-user gaining conventional PI space, with its own BGP advertised prefixes, the needs of all these end-users are satisfied with the advertisement of only the one MAB burdening the control plane of the BGP network.

Within each MAB, an end-user network will have a User Address Block (UAB) - an integer number of contiguous IPv4 addresses, or IPV6 /64 prefixes. End-users can divide this into as many “micronets” as they wish, where a micronet is one or more contiguous IPv4 addresses (or /64s) which are “mapped” to the same ETR address.

The mapping of each micronet, and the creation of these micronets within a UAB, is achieved by the end-user (or some device or organisation with their credentials) authenticating themselves and giving a mapping change command by a variety of methods, directly to a Root Update Authorisation System (RUAS) or indirectly, via one or more Update Authorisation Systems (UASes). Multiple RUASes work together to create a unified stream of mapping updates which are fanned out to full database ITRs (ITRDs) and full database Query Servers (QSDs) all over the Net, ideally within a few seconds. In contrast to some other proposals, the control of mapping is completely separated from ETRs. It is controlled by a unified, distributed, global system with no single point of failure.

Ideally, the Ivip approach to managing address space will prove so attractive to end-users of all types, that it will become widely and perhaps ubiquitously adopted. In time, this means that conventional space (RLOC space, in LISP terminology) may become a relatively small proportion of the total available space, and will be used by ISPs, including especially for ETRs as well as for space temporarily given to customers (including millions of home and SOHO DSL etc, customers) as PA (Provider Assigned) prefixes.

Benefits of SPI Space

SPI space will be attractive to end-users because it can be used for multihoming and portability (selecting any ISP or ISPs for Internet connectivity, whilst retaining the same addresses for hosts and the network) - with much lower costs and complications than the only current method of achieving this: an assignment of a BGP advertised prefix of PI space. Because Ivip uses a fast push system to control ITRs, the new type of space is well suited to a new approach to mobility - with generally optimal path lengths, few changes the mobile node’s software, and no changes to the correspondent node, for both IPv4 and IPv6.

In contrast to other proposals, Ivip’s mapping database and ITRs do not explicitly provide the inbound Traffic Engineering (TE) function of load-spreading. Every micronet is mapped to a single ETR’s address. Load sharing can be achieved as long as the load is spread over multiple IP addresses (or IPv6 /64s), by making each one a separate micronet, and mapping each micronet to a different one of several ETRs. Despite the absence of explicit ITR-based load sharing, this approach may be more fruitful than those of other proposals with explicit load sharing but slow mapping distribution systems, due to the low latency control (four to five seconds, ideally) which Ivip will provide over which ETR traffic is tunneled to.

Flexible Hybrid Push-Pull Architecture

Before discussing Ivip’s hybrid fast push-pull mapping distribution system, some alternatives will be discussed: LISP-ALT, TRRP, LISP-NERD and APT.

The global nature of the ALT query server system means that “initial packets” (packets in a new communication session which the ITR does not yet have mapping for) will frequently be delayed by times such as a second or more, causing significant problems for hosts and users at each end of communication sessions. TRRP also uses a pure pull mapping system, with similar delay problems. However, LISP-ALT’s delays are exacerbated by the structure of the ALT network, and the long geographic paths which initial packets will typically take across it. (See the page of LISP critiques for further details. <http://www.firstpr.com.au/ip/ivip/lisp-links/>) There are also difficulties achieving the highly aggregated structure of the ALT network whilst ensuring it is robust in the event of router or link failure.

The primary attraction of pure pull is that mapping information and changes are only sent where they are needed, and only a small subset of the total mapping database needs to be stored in any one ITR. This means LISP-ALT has no scaling limit in terms of the number of EID prefixes it handles (the equivalent of Ipvip's micronets). However, it seems unlikely that there will ever need to be more than about 10 million multihomed fixed end-user networks (one per 1000 people). LISP-NERD and Ipvip will have no difficulty scaling to such numbers. So the superiority of LISP-ALT depends on it being useful for hundreds of millions or billions of EIDS, which could only occur when it serves mobile devices such as cell-phones as the "end-user networks". No such applicability of LISP-ALT has been demonstrated, but it could work with the TTT Mobility architecture, as long as there was a relaxation of the current assumption in LISP that the ETRs are located at the end-user network.

TRRP resembles LISP-ALT in that it also uses a full-pull, global query server network for mapping distribution. Like LISP-ALT it can scale to arbitrary numbers of EIDs. Again, this will only be required if it is applied to mobile devices - and there has so far been no discussion of this occurring.

A pure push system such as LISP-NERD has only full-database ITRs. Every ITR gets a full feed of mapping updates, in the case of NERD by a relatively slow ITR-initiated file download system. The primary benefit of this is that all traffic packets are tunneled without delay to the correct ETR. The primary objection to pure push are that every mapping change is sent to, and stored at, every ITR in the Net. This would present efficiency and scaling problems if the map-encap scheme had to handle tens of millions, or billions, EID prefixes.

In summary, full-pull, global query server systems such as LISP-APT and TRRP are in principle able to scale to very large numbers of EIDs, they suffer from initial packet delays and from the reliability limitations inherent in a global query server network.

APT is a hybrid push-pull architecture, in which each participating ISP maintains several Default Mappers - which combine the functions of a full-database Query Server (QSD) and some functions which are performed by ITRs in LISP and Ipvip. Default Mappers must receive the full feed of mapping updates. The remaining ITRs are caching ITRs, which gain mapping information quickly and reliably from the Default Mappers in the same ISP network.

Ivip provides a more flexible hybrid push-pull approach than APT's. Full database Query Servers (QSDs) can be located anywhere. All ITRs are caching ITRs. However, an ITR connected directly to, or integrated within, a QSD is arguably a full-database ITR.

ITRs can be located in the DFZ, where they advertise one, several or perhaps all MAB prefixes and so attract packets sent from networks which have no ITRs. This OITRD (Open ITR in the DFZ - formerly "anycast ITRs in the core") approach enables Ipvip to be incrementally deployed without disruption. The same technique was adopted by LISP, with the same benefits - where it is called "Proxy Tunnel Routers".

Ivip enables local operators to choose where to place ITRs and QSDs - and therefore how deep into their networks the full feed of mapping updates needs to be pushed. Over the years to come, as technology changes, mapping updates grow, traffic patterns change, etc., Ipvip's flexibility in this regard should ensure that operators will be free to choose how much push and how much pull is used in their networks, and over what distances.

ITRs gain mapping information within tens of milliseconds, reliably, and with little cost in terms of query and response traffic, from local QSDs. Ipvip also provides optional caching Query Servers (QSCs) so that an ITR might gain mapping information from a nearby, low-cost QSC which already has the mapping information, or which obtains it from a local QSD, perhaps via one or more QSCs.

The rapid, reliable response from a local Query Server means that ITRs can buffer packets awaiting the response, and that the delay times will generally be short enough not to significantly impact end-users or higher level application protocols. This ability to flexibly locate low-cost ITRs is intended to enhance the operators' ability to reduce path lengths and to spread the load between many ITRs.

A further option is to integrate an ITR function inside a sending “host” ITFH (ITR Function in Host) - as long as it is not behind NAT. (ITRs and ETRs are all on stable public addresses. ETRs must be on BGP-managed conventional addresses and ITRs can also be on these or on Ipvip-mapped addresses.) Many hosts such as servers and desktop machines have abundant CPU and memory resources, and these can be used without significant cost to avoid the need for outgoing packets to be handled by purpose-built ITRs. If the CPU of a DSL modem/NAT/router is considered a “host”, then an example of ITFH would be adding an ITRC function to its software, using the DSL link’s IP address. ITFHs are not intended to be on addresses behind NAT, since they need to be directly reachable by the QSD or QSC from which they request mapping information. (Conceivably, ITRH hosts could be behind NAT, but this would require a special arrangement with their QSD or QSC which is not contemplated in the current Ipvip proposal.)

To request mapping information, ITRs send a single UDP packet, with a nonce (unique random number for this request packet) to a nearby QSD or QSC. The request contains a single IPv4 address (or for IPv6, the 64 most significant bits of a /64 prefix) which the ITR needs to gain the mapping for.

Query Servers send mapping responses to the requester (an ITR or perhaps a QSC) specifying the micronet which the request’s IP address (or /64) matches. For instance, the ITR requests mapping for 12.34.56.78 and receives mapping for the micronet 12.34.56.74 to 12.34.56.79. The mapping reply contains this starting address and length, and the mapping itself: a single ETR address to which the ITR should tunnel all packets whose destination address matches this micronet.

Each mapping response also has a caching time set by the Query Server, such as 10 minutes. Should the mapping for the micronet change (meaning a mapping update is received for this micronet by a QSD), the QSD *notifies* whatever device (an ITR, ITRF or QSC) requested the mapping, passing on the new mapping information and a new cache time. QSCs forward this to downstream devices in a similar fashion. In the event that the new mapping involves a change in how the SPI space is divided into micronets, the reply contains the new micronet’s starting address and length - the micronet which matches the originally requested IP address or /64. Caching time of updates will typically be however much of the original response’s caching time remains. This ensures that if a micronet’s mapping is frequently updated, the Query Server will not keep sending updates to the requester for any longer than the caching time of the initial response.

This *notification* arrangement (AKA “ITR cache update”) is a local-scale, real-time form of push, quite separate from the global fast push network, and only for micronets which cover an IP address (or IPv6 /64) for which the mapping has recently been requested. In this way, end-users can control the mapping used by all ITRs - all over the Net, ideally in a few seconds.

The mapping request from the ITR to the Query Server could be a UDP packet and the initial response from the Query Server could be a UDP packet secured by the nonce the ITR sent as part of its request. The cache update packet could be a UDP packet also secured with the nonce of the initial request, but this would require some kind of acknowledgement so the Query Server could ensure the ITR received the update. The exact mechanisms are for future development. Perhaps the easiest approach would be to have ITRs maintain a TCP session with its upstream Query Server and for responses and updates to be secured by the nonce of the original request. In practice, an ITR would be configured with the addresses of multiple Query Servers. Automated (zero ITR configuration) methods of discovering Query Server addresses will also be developed.

Fast Push Database Distribution

Some of the most novel and important aspects of Ipvip are the RUAS, Launch system and Replicator systems, which together enable end-user commands to be fanned out to hundreds of thousands or millions of ITRDs and QSDs in a few seconds.

The Launch system is a widely distributed set of servers, perhaps 8 in number, run collectively by the RUASes which manage the MABs of Ipv6-mapped address space. They work together, for instance on a one-second cycle time, to receive all mapping updates from all RUASes, establish a “quorum” of Launch servers which received all the same information, and then to launch a series of update streams of packets, each stream containing identical contents, from each Launch server, to the more numerous (perhaps 32) level 1 Replicators. (The details of how this will be done are still to be developed, but will employ established protocols which enable a distributed network of servers to make such decisions and so to operate robustly as a unified system.)

Replicators are conventional low-cost servers, with one or more gigabit Ethernet ports and special software. They will typically be located at major Internet exchanges where multiple physical long-distance links converge. Each Replicator, every second, receives two separate sets of update packets, with identical payloads. These come from two “upstream” replicators, ideally over different links, to help ensure physical redundancy in the event of an outage. These packets will be sent over an encrypted link (probably using RFC 4347 Datagram TLS), which prevents spoofing. The mapping data may be signed by the RUAS which generated it.

Each packet contains a sequence number at the start of its payload. If a packet with a particular sequence number is lost from one stream, it is unlikely that the same sequence numbered packet will be lost from the other stream. The Replicator immediately fans out the contents of the first received packet of each sequentially numbered packet, to some number N other Replicators on the next level. For N=20, each Replicator therefore consumes two streams from upstream replicators and sends 20 downstream - an amplification factor of $N/2 = 10$. In the early days of introduction, with low update rates, N could probably be much higher than 20, so the entire Replicator network could generate the requisite number of streams for the world’s QSDs with relatively few Replicators. Yet the system can grow incrementally to handle much larger numbers of QSDs and large update volumes.

The end result is that each QSD ideally receives two separate streams of real-time mapping information from two topologically separated Replicators. This should prove to be very robust against packet loss and reasonably robust against link failure. The system also involves QSDs downloading (from nearby, regional or perhaps distant HTTP servers) snapshots of each MAB’s mapping database at boot time, or after a disruption in mapping updates. Any packet which is missing from both streams can easily be detected by the QSD, and can be individually downloaded from the servers within a second or two.

Any push system is subject to the critique that one end-user’s mapping change burdens hundreds of thousands or millions of other devices (QSDs), most of which do not need the information. With Ipv6, mapping changes, or at least frequent mapping changes, should be charged for, so that end-user networks who generate the mapping changes contribute to the cost of running the global push system, and are inhibited from making changes and defining micronets in ways which impose costs on others which they are not prepared to pay for themselves. The fast nature of the push scheme enables mobility and other benefits for end-users, so they should be relatively comfortable paying a fee - such as a fraction of a cents to ten cents - per update.

The actual source of mapping changes

From the point of view of an ITR (including an ITFH in a sending host) the authoritative source of mapping information is its one active local Query Server. (An ITR will have alternatives to use if the current Query Server fails to respond.)

If that Query Server is a QSC, its authoritative source of mapping is its one upstream Query Server. (Likewise, QSCs will have multiple upstream Query Servers to use if the current one fails to respond.) There may be multiple levels of QSCs - but there is no requirement that QSCs be used at all.

Directly, or indirectly via one or more QSCs, an ITR’s authoritative source of mapping information is a nearby QSD.

From the point of view of the QSD, the authoritative sources of mapping are the multiple RUAS systems. Each RUAS is responsible for one or most likely hundreds or thousands of MABs. Each MAB has its own stream of mapping updates and its own snapshots for QSDs to download at boot time, or to re-establish sync after a disruption. Normally, QSDs gain this mapping for each MAB via an initial downloaded (pull, initiated by the QSD, from one of multiple distant servers) snapshot, which is then updated, potentially every second, by the real-time stream of mapping updates.

The source of those updates is normally the two (typically) upstream Replicators, each of which normally sends a full stream of updates to the QSD. In the event that one or more packets are missing from one Replicator, these will typically arrive from the other. In the event that one or more packets are missing from both Replicators (and a QSD could have feeds from one, two, three or any number of Replicators) then the QSD will need to initiate a download from a distant server of the missing packet. In normal operation, only the packets from the Replicators are needed for the QSD to remain entirely up-to-date with the mapping of the micronets in each MAB. The mapping update stream will contain, at regular intervals, hashes of each MAB's mapping so the QSD can check the validity of its copy of that MAB's mapping. If there is a discrepancy, then the QSD can initiate a download of an entire snapshot which matches that hash. (There is the potential for using hierarchical trees of hashes to narrow down the area where data is missing and so enable the downloading of a subset of the MAB's snapshot.)

So from the QSD's point of view, the authoritative source of mapping for any one MAB is the RUAS which is responsible for that MAB.

The RUAS may directly rent the space of a given MAB to multiple end-user networks. Alternatively it may operate the MAB for some other company who rents the space to end-user networks. (A third alternative is for a large end-user network to have its own MAB, perhaps made from an already existing PI prefix, which it contracts an RUAS to handle the mapping for.)

From the RUAS's perspective the mapping for a given MAB, including how the MAB is divided into micronets, is controlled by end-user networks whose micronets these are. If the MAB belongs to another organisation, the RUAS receives a single real-time stream of mapping change commands from that organisation - which may itself receive mapping changes from its multiple end-user network customers.

Administratively, each end-user network rents a UAB (User Address Block) of SPI space from some organisation - either an RUAS directly or from some other organisation who uses an RUAS to send the mapping for this MAB to all QSDs via the Launch servers and the global Replicator system.

It is up to the end-user network how the UAB is broken into micronets. A UAB can be any number of IP addresses (or IPv6 /64s) with an arbitrary starting point. (That is, neither the UABs or micronets need be "prefixes" in the traditional IP fashion.) The end-user network can change the way the UAB is subdivided into micronets at any time. They can change the mapping for each micronet (a single ETR address) at any time. Each such change involves a fee, so the RUAS gains revenue for running its share of the Launch and Replicator systems.

Technically, it is possible for the end-user network to directly control the mapping. For instance, this could be done manually via a secure web-interface directly to the RUAS or to whatever company the end-user network rents their SPI space from. This would enable manual responses to outages so that a multihomed network could continue operating via one ETR at one ISP after another ISP and its ETR became unusable.

A more likely arrangement for most multihomed end-user networks would be for them to contract the services of a specialised "Multihoming Monitoring Company" (MMC). The MMC would technically control the mapping, by the end-user network first establishing a username and password for the MMC to control the mapping of its UAB. This would enable the MMC's distributed system of servers to directly and automatically send mapping changes to the RUAS or to whatever organisation accepts these mapping changes and sends them to the RUAS.

The MMC's servers would be configured to continually probe reachability of the end-user network via its two or more ETRs. In the event of the network being unreachable via ETR-A, the MMC would issue map update commands within seconds so that all ITRs would tunnel packets addressed to the end-user network's micronets via ETR-B. Exactly how this probing occurs, and how the decisions are made about changing mapping in response to a detected outage, are entirely a matter between the end-user network and its chosen MMC. There may well be IETF standards in this field, but they are not a part of the Ipvip proposal. In this way, the end-user network can choose exactly how reachability is probed, how the multihoming service restoration decisions will be made, and how the mapping will be changed once it appears that the outage has been resolved.

Administratively, the authoritative source of mapping is the end-user network, but that network has delegated responsibility to the MMC of its choice.

If the end-user network was also using Ipvip to load-balance incoming traffic over two or more ISPs and their respective ETRs, then the MMC system needs an additional elaboration. The MMC's purpose is firstly to ensure service continuity for all the micronets in the event of one ETR (or its ISP or the link to that ISP) failing: as described above. As long as the end-user network can be reached by both ETRs, then the MMC's second responsibility is to balance the load as suits the end-user network. As long as both ETRs are operating, then the MMC needs to accept, via some secure mechanism, mapping update commands from the end-user network itself. These may be any mix of manual and/or automatic changes. Routers, servers and/or manual operators at the end-user network are the only possible source of mapping updates for the purpose of load balancing. So as long as both ETRs are up, the MMC system regards the appropriate sources of mapping updates in the end-user network as the authoritative source of mapping.

In the TTR Mobility model, the end-user network owner (such as an individual with a cellphone) contracts with a TTR company for the use of its global TTR network. That company will also be given the credentials it needs to control the mapping of the end-user network's micronets - probably just a single IPv4 address or IPv6 /64.

Benefits of Fast Push

Fast push provides Ipvip with a number of unique benefits which enables the new kind of address space to be used for a new global approach to Mobility, and which enables considerable simplification of ITR and ETR functionality compared to pull or slow push schemes. A fuller statement of these benefits can be found in *ivip-db-fast-push*. These include:

- **Modular separation of the multihoming restoration functions.** Since end-users can control the mapping used by all ITRs with a few seconds latency, there is no need to have the ITRs make decisions about which of multiple ETRs to tunnel packets to. Ipvip does not monolithically integrate the multihoming monitoring tasks or the decision making which follows, into the map-encap scheme, as is the case with LISP, APT and TRRP. Instead, the end-user chooses their own system of monitoring and decision making, and uses that to drive the Ipvip system. This distinction makes Ipvip - or any other scheme with similar modularity - a better choice for IETF development than those which integrate the following functions monolithically:
 - The exact methods by which each ETR's reachability could be determined, presumably by each ITR (or APT Default Mapper) operating in isolation.
 - Similarly, any other reachability functions, such as determining whether an ETR is capable of delivering packets to the destination network.
 - The logic of all decisions regarding ensuring continued connectivity for multihomed networks, and likewise for TE. These need to be codified as part of the specification of LISP, APT, TRRP etc, because they need to be part of the functional specification for all ITRs and for the format of mapping information.

- Since these functions involve ITRs probing ETRs, it is also necessary for the scheme to build additional functionality into the ETRs.
- **Reduction in the size of the mapping information.** Only a single ETR address needs be sent to the ITR, reducing the size of updates and the storage requirements of QSDs. (However, more updates may need to be sent.)
- **Reduced ITR and ETR functionality.** As noted above, Ipv ITRs and ETRs have a less complex job to do, and so are easier to implement, manage and operate securely.
- **Greater security through simplification and modularization.** The centralised, unidirectional, fast-push scheme is probably easier to secure against attack, packet loss and link failure than a global query server network such as LISP-ALT or TRRP.
- **IPv4 and IPv6 mobility with generally optimal path lengths.** As discussed below.

Outer Source Address = Sending Host's Address

This section applies only to encapsulation, not to the Modified Header Forwarding (MHF) approaches. With MHF, existing ISP filtering arrangements for the source address of packets entering their networks from the DFZ will operate normally.

Other map-encap proposals (LISP, APT and TRRP) tunnel packets from the ITR to the ETR with the outer header's source address set to the ITR's address. Ipvip's approach is less conventional: the outer source address is that of the original sending host. This has a disadvantage in that ETRs cannot tell which ITR sent the packet, but there are two important advantages:

- If an ISP network's border routers filter incoming packets to drop any with a source address matching one of the network's prefixes (meaning the packet has a spoofed source address) then the map-encap scheme should be capable of enforcing similar filtering on the decapsulation operations of all ETRs in that network. Brute force replication of this filtering in each ETR would be inordinately expensive. Ipvip enables this filtering to be extended to all decapsulated packets which were tunneled to the ETR from outside the network by the simple method of the ETR dropping any decapsulated inner packet whose source address does not match the source address of its outer header. Packets tunneled from ITRs inside the network are decapsulated and forwarded normally, since their outer headers have not been subject to filtering at the border routers.
- With modest software modifications, a traceroute program running on the sending host will be able to recognise the ICMP messages generated by routers in the ITR to ETR tunnel.

IPTM - Integrated PMTUD and Fragmentation Management

This section applies only to encapsulation - not to the use of Modified Header Forwarding.

In any map-encap scheme, the ITR's encapsulation headers increase the likelihood of the resulting packet exceeding the MTU limits of one or more routers in the ITR to ETR tunnel, whilst also preventing any Packet Too Big (PTB) packets generated by those routers from causing the Sending Host (SH) to adjust the lengths of its packets accordingly. The ITR-ETR tunnel is unusual compared to other tunnel scenarios in that it is an ad-hoc, fleeting, arrangement in which overhead for managing Path Maximum Transmission Unit Discovery must be minimised as much as possible, while ensuring reliable delivery of application data.

In April 2008, I devised a new and promising approach for handling these challenges, including the need for the ITR to somehow cause RFC 1191 compatible hosts to send packets of a carefully chosen size - so that once encapsulated, they just fit within the tunnel's MTU limits. Please see <http://www.firstpr.com.au/ip/ivip/pmtud-frag/> for a fuller explanation of this new "IPTM: ITR Probes Tunnel MTU" approach, which does not involve any extra headers beyond Ivip's minimal IP-in-IP encapsulation for most traffic packets. For each ETR, an ITR develops a reliable estimate of the PMTU to that ETR, by using traffic packets as probes.

IPTM can work with RFC 4821 Packetization Layer PMTUD, but does not require this. It assumes the existence of RFC 1191 PMTUD in the sending host. IPTM is not intended to handle DF=0 (IPv4) packets beyond a certain length. See the abovementioned page for more details. It is intended to efficiently handle DF=1 packets of any length, including when most DFZ paths have an MTU of ~1500, and when some or all of them have jumboframe MTUs such as ~9000. There appears to be no other attempt (such as within LISP, APT or TRRP) to smoothly transition between the current ~1500 regime and ~9000 or beyond.

Two variables are maintained, one initially at a very high value and one initially set to the MTU of the next hop towards the ETR. These two values delineate the range of packet lengths for which the ITR is not sure whether this length exceeds the PMTU. As successive packets within this range (once encapsulated) are sent, special protocols are used by which the traffic packet is either delivered and the ITR is able to verify this - or it is not delivered and the ITR reliably detects the non-delivery. Consequently, with each such packet whose length is within the "zone of uncertainty" the ITR is able to narrow the zone, adjusting either the upper or the lower value towards the other, eventually causing them to converge to the same value. Then, the ITR has a reliable estimate of the PMTU, and will reject packets with a PTB message to the sending host, if their length, plus the encapsulation overhead, would make them longer than the PMTU for this ETR.

The probing of the PMTU is only done for packets whose length, once encapsulated, lies between these two values. The probing involves two packets and a special protocol between the ITR and ETR. This maintains the ability of the ISP source address filtering system to work, and results in either the delivery of the packet or the ITR discovering reliably that it was not delivered.

IPv4 and IPv6 Mobility

Space permits only a terse description of this capability. This does not place any burdens on the architectural complexity of the basic Ivip system. Please refer to the paper:

TTR Mobility Extensions for Core-Edge Separation Solutions to the Internet's Routing Scaling Problem

Robin Whittle, First Principles, Rosanna, Vic, Australia
Steven Russert Boeing Phantom Works, Seattle, WA, USA
2008-08-25 <http://www.firstpr.com.au/ip/ivip/#mobile> .

This approach to global mobility is quite different from traditional Mobile IP. However, it can work well with mobile hosts which use traditional Mobile IP.

The Mobile Node (MN) retains its Ivip-mapped IP address space (including perhaps a micronet of one IPv4 address or a /64) wherever it is located, and establishes one or more Care-of Addresses (CoAs) in various networks. These may be fixed IP addresses, DHCP addresses, addresses behind NAT, or Mobile IP addresses.

For instance, a laptop or cellphone may have a WiFi connection to ISP A and so a temporary CoA (perhaps behind NAT) in that network. It then establishes a link via 3G to ISP B, and so gains another CoA in that network. The MN needs to establish tunnels from each such CoA to one or more ETR-like devices, which are optimised for mobility. These Translating Tunnel Routers (TTRs) combine ITR and

ETR functions with the ability to authorise and service a two-way encrypted tunnel established from the mobile device. An external, distributed system of servers enables the MN's software to choose TTRs which are either within, or close to, the access network it is currently connected to. The TTRs and the TTR location systems would typically be operated by companies who charge end-users.

The MN sends outgoing packets to the TTRs, which are able to forward them to the rest of the Internet, perhaps performing ITR encapsulation (or "Forwarding") at that point. The MN and/or some external system controls the mapping of the micronet for this device's address space, causing all the world's ITRs to tunnel traffic packets to one of the one, two or more TTRs to which the device has connections. (Load sharing over two TTRs could be achieved with two micronets.)

Assuming the TTRs are relatively close to each point of connection to the separate networks, then total path lengths from corresponding hosts will generally be optimal or close to optimal. There is no "home agent" or "triangle routing". The system should work fine with both IPv4 and IPv6, with no changes required for correspondent hosts (the hosts the MN is communicating with), and only some additional software, rather than actual host stack changes, for the mobile host. The mapping of the end-user's micronet is done when their MN uses a new TTR. Selecting a closer TTR to the currently used access network is not required to maintain connectivity, but is desirable to reduce total path lengths. Typical end-users would not need a new TTR as long as their access networks had border routers within about 1000km of the current TTR.

So this approach to mobility does not require a mapping change every time the MN acquires a new CoA - only when the new CoA is so far from the current TTR that it is worth establishing a tunnel to a closer TTR and then making a mapping change to direct packets to that new TTR.

Modularity and Extensibility

Any new architectural addition for the Internet should be as elegant, powerful, modular and extensible as possible. By separating out multihoming monitoring and decision making and making these the responsibility and choice of end-users, the Ivip system is a conceptually simpler (compared to LISP, APT and TRRP), modular, global tool for catching packets addressed to particular BGP advertised prefixes, and tunneling them to an ETR chosen, effectively in real-time, by the end-user whose micronet the packets are addressed to. Perhaps in the future there will be a need for some more sophisticated functions, or a completely novel and valuable use of the tunneling (encapsulation or MHF) functionality for purposes different from multihoming, portability, TE and mobility.

Perhaps in the future there will be a way of using the core-edge separation scheme to support a novel transition mechanism from IPv4 to IPv6, or to some other system. A fast-responding, global network of ITRs and ETRs, with a secure, unified yet decentralised fast-push mapping distribution system, would be a far better basis for new architectural developments than a system such as LISP, which bundles together a large, but functionally limited, set of mechanisms to solve the scaling problem as we conceive of it today.

Analysis wrt RADIR Problem Statement

The RADIR Problem Statement *draft-narten-radir-problem-statement-01* discusses the BGP routing scaling problem in terms of general principles regarding costs and benefits being aligned so one party's actions should not unnecessarily burden other parties who derive no benefit from those actions. I quote selected pieces of the Problem Statement and comment on how Ivip relates to these.

“It is desirable that the routing and addressing system exert the least possible back pressure on end user applications and deployment scenarios, to enable the broadest possible use of the Internet.”

Ivip enables increasing amounts of address space to be managed by a new lightweight mechanism which produces no incremental burden on the BGP control plane beyond the advertisement of the MAB prefix. The new kind of space can be divided with arbitrary finesse and is suitable for end-users of all sizes who need multihoming, portability, TE and mobility. To the extent that Ivip becomes part of the routing and addressing system, there will be a great reduction in costs and other barriers relating to large numbers of end-users using the Internet efficiently.

“Misalignment of cost and benefit” with today's BGP system.

Ivip will place minimal load on the BGP control plane, considering the number of end-users it will support. However, the push nature of the mapping system opens a similar problem regarding the burden of each end-user's mapping changes. This is expected to be much lighter than with BGP, since the push system is streamlined and purpose built for propagating these small update messages. Nonetheless, some kind of payment arrangements will be necessary so the end-users who benefit from the mapping change pay for the cost of the system.

DFZ Routing Table Size

To the extent that end-users with PI space today - or of the future who would otherwise gain PI space later - adopt Ivip micronets, and to the extent that many such end-users' needs can be accommodated in a relatively small number of MABs, Ivip will reduce the absolute number of DFZ routes and/or greatly reduce future growth in these routes.

The Summary is a list of criteria, similar to design goals, for a new approach to routing and addressing:

1. Provides sufficient benefits to the party bearing the costs of deploying and maintaining the technology to recover the cost for doing so.

Organisations such as ISPs who run RUASes and contribute to the cost of the Launch system and upper parts of the Replicator system will be able to charge for the use of the address space, according to number of IP addresses, number of micronets, and number of updates sent to the system. The lower (much more numerous) parts of the Replicator system are likely to be run by ISPs, as will be the ITRs and Qs in each ISP network. ETR costs will likewise be borne by customers in ISP networks. The cost of OITRDs the DFZ will be borne by the RUASes, and through them by their customers who rent address space and pay for their share of OITRD traffic.

2. Reduces the growth rate of the DFZ control plane load. ...
... Any change to the control plane architecture must result in a reduction in the overall control plane load, and shouldn't simply shift the load from one place in the system to another, without reducing the overall load as a whole.

For a given number of multihomed end-users, Ivip greatly reduces the load on the BGP control plane, and moves the load to a different and generally more efficient set of processes in the Ivip system. This will be purpose built to handle these loads in a scalable manner, and to allow

charging for updates and other activities which would otherwise place a burden on parts of the system which derive no direct benefit from these activities.

3. Allows any end site wishing to multihome to do so.

Ivip provides this.

4. Supports ISP and enterprise TE needs.

Ivip's TE does not allow load-spreading for a single micronet, but provides potentially rapid control of small and dynamically created micronets. Ivip mapping is controlled by end-users and not by ISPs (unless the end-user wishes the ISP to control the mapping of their micronets). For a given ETR, an ISP with multiple border routers can advertise that ETR's conventional RLOC address space on any such router, continuing the current practice of BGP-based TE at the level of advertised prefixes.

5. Allows end sites to switch providers while minimizing configuration changes to internal end site devices.

Ivip provides this because the new type of address space is entirely portable, can be of any size, and has no direct impact on the BGP control plane other than the requirement that each MAB be advertised in BGP. However, each MAB would typically encompass thousands or millions of individual micronets.

6. Provides end-to-end convergence/restoration of service at least comparable to that provided by the current architecture

Ivip has no inbuilt multihoming restoration functions. However it is reasonable to expect these to be available in great variety outside the Ivip system, and for end-users to choose whichever of these systems suit their needs. This is a more flexible, modular, open-ended solution than is possible if the multihoming monitoring and service restoration functions were built in to the map-encap scheme.

Analysis wrt RRG Design Goals

The RRG Design Goals <http://tools.ietf.org/id/rrg> are listed with R = Required, SD = Strongly Desired, D = Desired.

- 3.1 R Improved routing scalability
Yes.
- 3.2 SD Scalable support for traffic engineering
Yes.
- 3.3 SD Scalable support for multi-homing
Yes.
- 3.4 D Scalable support for mobility
Yes - Ipvip's "real-time" control of mapping best supports the TTR approach to mobility, as described above and at <http://www.firstpr.com.au/ip/ivip/#mobile>.
- 3.5 SD Simplified renumbering
Actually "It is strongly desired that a new architecture allow end-sites to change providers with significantly less disruption.". With Ipvip, changing providers will involve some effort and potential for disruption, but none of this will be due to renumbering, since the end-user's address space remains the same and is completely portable.
- 3.6 D Decoupling location and identification
Ivip achieves "decoupling of host location and identification information" by the end-point being identified by a micronet address, and its physical location, at least in terms of the ETR it uses, being specified by a conventional RLOC address. This is all part of the solution to the scalable routing problem, not a separate or incompatible process.
- 3.7 SD First-class elements
The new type of address space could be considered a "first class element" if the specific restrictions it involves were deemed acceptable within this definition, including: not to be used for ETR addresses, for any critical part of the Ipvip system or for ISPs who sell connectivity (although it probably could be used for the latter in many circumstances).
- 3.8 SD Routing quality
Longer path lengths (stretch) are possible, depending on the location of the ITR and ETR with respect to the path which a packet from source to destination would otherwise flow. However, stretch will typically be zero or minimal, assuming widespread deployment of ITRs, including OITRDs. There are no obvious sources of instability in Ipvip, or any sense of "convergence", because it is not a traditional self-organising routing system. It is simply a worldwide network of ITRs controlled in real-time by end-users, which tunnel packets to a given ETR depending on which micronet the destination address is within.
- 3.9 R Routing security
Any additional constructs are likely to degrade security, rather than improve it. A major requirement for any map-encap scheme is to make it robust against attack, packet loss and link failure in its control plane. Much work remains to be done, so the answer to this can only be given with a thorough assessment or trial of the nearly finished Ipvip system.
- 3.10 R Incremental Deployability <http://psg.com/lists/rrg/2008/msg00957.html>
Ivip address space has immediate benefits for the end-users who are assigned it, even in the earliest stages of adoption. As long as there are sufficient OITRDs, sending hosts in networks which lack ITRs should have sufficiently reliable and low stretch connectivity to the destination hosts that there will be few problems to deter adoption.

Modified Header Forwarding (MHF) instead of encapsulation

ETR Address Forwarding (EAF) - for IPv4

This is fully described in I-D *draft-whittle-ivip4-etr-addr-forw-01* (work in progress, 2008-08-22). This is conceptually and technically quite simple. Prefix Label Forwarding (PLF) - for IPv6 - is somewhat more complex.

The ITR modifies the IPv4 header to contain a 30 bits of the ETR address. The new packet format is denoted by the “Evil Bit” (bit 48) to 1. The 30 bits are located in the bits currently used for the More Fragments bit, the Fragment Offset and the Checksum. ITRs do not accept fragmented packets, or fragmentable packets longer than some globally agreed constant, which would be somewhat below 1500 bytes. (Fragmentable packets which are accepted are converted to non-fragmentable packets by the ETR, which raises some potential problems if there is a PMTU restriction below this figure between the ETR and the destination host. Further work is required on this, but the workaround is to set the value to some value low enough that all end-user networks adopting Ipvip can avoid such restrictions without undue inconvenience.)

This “somewhat below 1500” value would be some globally agreed PMTU value which could be assured between all ITRs and ETRs. Identical restrictions regarding the ITR dropping DF=0 packets beyond such a length would apply to a map-encap scheme which properly handled PMTUD problems, as described above for the IPTM approach to handling the PMTUD problems inherent in encapsulation.

The advantages of EAF over encapsulation include:

1. There is no transmission overhead - the packet is not made any longer.
2. Conventional RFC 1191 PMTUD is supported over the entire path, including between the ITR and ETR, without any involvement of the ITR.
3. Traceroute is expected to work over the entire path.

ETRs would be located on every fourth IP address. This is not expected to be a significant restriction.

In principle, all routers in the DFZ need to be upgraded to forward packets with bit 48 = 1 according to this 30 bit address, rather than the destination address. This is on the assumption that ITRs and ETRs are at the borders of ISP networks. In practice, it will also be necessary to upgrade internal routers between all ITRs (including any ITR functions in sending hosts) and the border routers, and likewise between the border routers and all ETRs.

In practice, it would not be necessary to upgrade every DFZ router. For instance, it would not be necessary to upgrade any DFZ router which only handled packets for one or more ISP network(s) and/or conventional (BGP managed, not Ipvip managed) end-user network(s) which did not contain either ITRs or ETRs.

The changes to routers can presumably be implemented as firmware changes in any router which uses firmware, rather than hardware, to classify and forward IPv4 packets. In the DFZ, and perhaps to a lesser extent in internal routers, by the time Ipvip is to be introduced globally, most or all such routers may be of this kind. Firmware upgrades involve some modest effort by the router vendor, and then some administrative work to upgrade the actual routers. No new hardware would be required, except when a non-upgradable router in the DFZ, or in an internal router location, needs to be replaced with one which has the upgraded forwarding capability.

The upgrade only affects the FIB section of the router. It has no impact on the RIB or on the BGP implementation. With EAF, the ETR function is very simple. It simply zeroes the 30 bits and the “Evil Bit”, which reconstitutes the initial state of the packet. The packet is then a non-fragmentable packet,

identical to the original, except where the original was fragmentable (DF=0). Then, just as with encapsulation, the ETR has its own methods of forwarding the packet to the correct end-user network, based on the packet's destination address.

Prefix Label Forwarding (PLF) - for IPv6

The IPv6 header does not have as many bits available as in the IPv4 header. 20 bits are available in the "Flow Label", which is not currently used. By redefining the "Flow Label", and implementing suitable changes in DFZ routers (and in internal routers between the border routers and any ITRs and ETRs), PLF could be introduced to the IPv6 Internet. Given the lower urgency of solving the IPv6 routing scaling problem (due to slow uptake of IPv6), it would be highly desirable if Ivip for IPv6 was deployed in this way, eliminating the need for encapsulation and the ITR and ETR complexities of handling the PMTUD problems caused by encapsulation.

This proposal is not yet documented in an I-D. Please see <http://www.firstpr.com.au/ip/ivip/ivip6/> for a fuller explanation. The advantages are the same as those listed for EAF above. However, PLF is a somewhat more complex, subtle and flexible arrangement than EAF.

It is not possible to build as many bits into the IPv6 header as would be required to directly specify an ETR address, which would be at least 64 bits, and ideally as many as 128 bits. 20 bits are available and the current proposal is to reserve 19 of them for use in the DFZ, with one final (MSB) bit being set to 0. When the MSB is set to 1, the 19 bits can be used inside each ISP network for whatever purposes the network operator chooses. This is likely to be a local system of forwarding similar to that described below for the DFZ.

The central premise of PLF is that it will always be sufficient, in the IPv6 Internet, to have an upper limit of 2^{19} on the number of ISP networks (or sites) which could contain an ETR. Each such ISP will have a prefix, within a special set of prefixes, by which its ETRs can be reached. ISPs are likely to have such a prefix for each of their physical sites.

The purpose of PLF is to enable the packet to be forwarded from the ITR to the border router of whichever ISP site contains the ETR. It is up to that border router how it forwards the packet to the ETR. This is trivial if the site has only a single ETR. For sites with multiple ETRs, it will be necessary to perform a mapping lookup on the destination address, similar to that done by the ITR, to determine the exact, full 128 bit, address of the ETR. How the ISP's border routers forward the packet to those ETRs is not currently specified as part of the Ivip proposal. However, it is possible that by implementing a similar system using the other 2^{19} bit combinations, that a reasonably efficient, internal, non-encapsulation, system can be used within the ISP network to forward the packets to ETRs.

There is a special set of 2^{19} contiguous prefixes, which are known as Core Egress Prefixes (CEPs). These may be used for any purpose, but all ETR addresses must be within one of these prefixes. For instance, the lowest CEP is number 0, at $E000:0000::/32$. The highest, number 524,287, is $E007:FFFF::/32$. ISPs advertise these conventionally in BGP.

DFZ routers (and all internal routers between ITRs and the border router) need to be upgraded so that when the 20 bits currently used for the Flow Label are non-zero, and the most significant bit (MSB) of these 20 bits is 0, the packet is forwarded according to the remaining 19 bits, which map directly to the corresponding CEP prefix as just described. (This means that the lowest CEP prefix, number 0, is not used.)

This will require changes to the router's FIB, and to the RIB, so that the FIB has the FEC (Forwarding Equivalence Class) information for each CEP prefix which is currently advertised. No changes are required to the BGP implementation.

For an ISP to use this system for its own internal forwarding system, its routers would need similar functionality to that just described. However, for the 2^{19} possible combinations of the old Flow Label bits, when the MSB = 1, each ISP would need to be able to configure their routers to map these to some ISP-chosen set of prefixes. These need not be a contiguous set, as is the most obvious solution for the DFZ arrangement.

With PLF, the overall process is as follows.

The ITR, including an ITR function in the sending host, looks up the mapping for the destination address and, in principle, receives a 128 bit ETR address. (In practice, it only needs a subset of this.) All ETR addresses (in the above example) are within $E000:0000::/13$. The ITR sets the 20 bits of the old Flow Label to 0 (for the MSB) with the other 19 bits being set to which /32 in the above /13 the ETR is located within. In principle, this means that ITRs only need 19 bits of mapping information, rather than a full 128 bit ETR address. This could somewhat simplify the mapping reply messages, and will certainly simplify the caching requirements in the ETR, making them less demanding than the 30 bits required for caching IPv4 ETR addresses, or the full 128 bit ETR address which is required for IPv6 using encapsulation.

The ITR now forwards the packet according to the 19 bits just set in the old Flow Label. This will typically cause the packet to be forwarded towards a border router. The border router, being a DFZ router, forwards the packet according to these 19 bits, as do other DFZ routers and the packet arrives at one of the border routers of the ISP site whose CEP prefix corresponds to this 19 bit value.

There, the border router clears the 20 bits, returning the packet to its original format. At this point, if there is a single ETR at that site, the packet is forwarded by some means to that ETR. If there are multiple ETRs, then the border router needs to perform a mapping lookup on the destination address, to determine the full ETR address, so the packet can be forwarded there. In practice, the border router would rarely need to perform such mapping lookups, since it would be constantly handling packets addressed to whatever end-user networks which use the ETRs in this ISP site - so it would already have this mapping in its cache.

Changelog

The filename of each entry indicates the name of the version in the <http://www.firstpr.com.au/ip/ivip/> directory

Ivip-summary-2008-02-19.pdf
Initial version.

Ivip-summary-2008-04-23.pdf (The date at the top of the document was mistakenly 2008-02-23).
Updated to use "OITRD" instead of "Anycast ITR in the core/DFZ" and to include the March 2008 approach to solving the PMTUD problems caused by encapsulation.

Ivip-summary-2008-12-21.pdf
Added explanation at the start and a section at the end about "Forwarding", which is now the long-term goal of Ivip, rather than the use of encapsulation. Linked to the paper written with Steve Russert about the mobility extensions to Ivip. Added this changelog.

Ivip-summary-2009-04-15.pdf
Current version with various minor improvements. Removed mention of Full Database ITRs - all are now Caching ITRs, so there is no need for the terms ITRD or ITRC. Added new section "The Actual Source of Mapping Changes".