

IPNGWG Working Group
Internet Draft
[draft-ietf-ipngwg-scoping-arch-03.txt](#)
November 2001
Expires May 2002

S. Deering
Cisco Systems
B. Haberman
No Affiliation
T. Jinmei
Toshiba
E. Nordmark
Sun Microsystems
A. Onoe
Sony
B. Zill
Microsoft

IPv6 Scoped Address Architecture

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Abstract

This document specifies the architectural characteristics, expected behavior, textual representation, and usage of IPv6 addresses of different scopes.

1. Introduction

Internet Protocol version 6 includes support for addresses of different "scope", that is, both global and non-global (e.g., link-local, site-local, etc.) addresses. While non-global addressing has been introduced operationally in the IPv4 Internet, both in the use

of private address space ("net 10", etc.) and with administratively scoped multicast addresses, the design of IPv6 formally incorporates

the notion of address scope into its base architecture. This document specifies the architectural characteristics, expected behavior, textual representation, and usage of IPv6 addresses of different scopes.

[2.](#) Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC 2119](#)].

[3.](#) Basic Terminology

The terms link, interface, node, host, and router are defined in [RFC 2460]. The definitions of unicast address scopes (link-local, site-local, and global) and multicast address scopes (interface-local, link-local, etc.) are contained in [[ADDRARCH](#)].

[4.](#) Address Scope

Every IPv6 address has a specific scope, that is, a topological span within which the address may be used as a unique identifier for an interface or set of interfaces. The scope of an address is encoded as part of the address, as specified in [[ADDRARCH](#)].

For unicast addresses, there are three defined scopes:

- o Link-local scope, for uniquely identifying interfaces within (i.e., attached to) a single link only.
- o Site-local scope, for uniquely identifying interfaces within a single site only. A "site" is, by intent, not rigorously defined, but is typically expected to cover a region of topology that belongs to a single organization and is located within a single geographic location, such as an office, an office complex, or a campus. A personal residence may be treated as a site (for example, when the residence obtains Internet access via a public Internet

service provider), or as a part of a site (for example, when the residence obtains Internet access via an employer's or school's site).

- o Global scope, for uniquely identifying interfaces anywhere in the Internet.

The IPv6 unicast loopback address, `::1`, is treated as having link-local scope within an imaginary link to which a virtual "loopback interface" is attached.

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

2

Internet Draft

IPv6 Scoped Address Architecture

November 2001

Anycast addresses [[ADDRARCH](#)] are allocated from the unicast address space and have the same scope properties as unicast addresses. All statements in this document regarding unicast apply equally to anycast.

For multicast addresses, there are fourteen possible scopes, ranging from interface-local to global (including both link-local and site-local). The interface-local scope spans a single interface only; a multicast address of interface-local scope is useful only for loopback delivery of multicasts within a single node, for example, as a form of inter-process communication within a computer. Unlike the unicast loopback address, interface-local multicast addresses may be assigned to any interface.

There is a size relationship among scopes:

- o for unicast scopes, link-local is a smaller scope than site-local, and site-local is a smaller scope than global.
- o for multicast scopes, scopes with lesser values in the "scop" subfield of the multicast address [[ADDRARCH](#), [section 2.7](#)] are smaller than scopes with greater values, with interface-local being the smallest and global being the largest.

However, two scopes of different size may cover the exact same region of topology. For example, a site may consist of a single link, in which both link-local and site-local scope effectively cover the same topological span.

[5](#). Scope Zones

A scope zone, or a simply a zone, is a connected region of topology of a given scope. For example, the set of links connected by routers within a particular site, and the interfaces attached to those links, comprise a single zone of site-local scope. Note that a zone is a particular instance of a topological region (e.g., Alice's site or Bob's site), whereas a scope is the size of a topological region (i.e., a site or a link or a ...).

The zone to which a particular non-global address pertains is not encoded in the address itself, but rather is determined by context, such as the interface from which it is sent or received. Thus, addresses of a given (non-global) scope may be re-used in different zones of that scope. For example, Alice's site and Bob's site may each contain a node with site-local address fec0::1.

Zones of the different scopes are instantiated as follows:

- o Each interface on a node comprises a single zone of interface-local scope (for multicast only).
- o Each link, and the interfaces attached to that link, comprises a single zone of link-local scope (for both unicast and multicast).
- o There is a single zone of global scope (for both unicast and multicast), comprising all the links and interfaces in the Internet.
- o The boundaries of zones of scope other than interface-local, link-local, and global must be defined and configured by network administrators. A site boundary serves as such for both unicast and multicast.

Zone boundaries are relatively static features, not changing in response to short-term changes in topology. Thus, the requirement that the topology within a zone be "connected" is intended to include links and interfaces that may be only occasionally connected. For example, a residential node or network that obtains Internet access by dial-up to an employer's site may be treated as part of the employer's site-local zone even when the dial-up link is disconnected. Similarly, a failure of a router, interface, or link that causes a zone to become partitioned does not split that zone into multiple zones; rather, the different partitions are still

considered to belong to the same zone.

Zones have the following additional properties:

- o Zone boundaries cut through nodes, not links. (Note that the global zone has no boundary, and the boundary of an interface-local zone encloses just a single interface.)
- o Zones of the same scope cannot overlap, i.e., they can have no links or interfaces in common.
- o A zone of a given scope (less than global) falls completely within zones of larger scope, i.e., a smaller scope zone cannot include more topology than any larger scope zone with which it shares any links or interfaces.
- o Each zone is required to be "convex" from a routing perspective, i.e., packets sent from one interface to any other interface in the same zone are never routed outside the zone.

Each interface belongs to exactly one zone of each possible scope.

[6. Zone Indices](#)

Considering the fact that the same non-global address may be in use in more than one zone of the same scope (e.g., the use of site-local address fec0::1 in both Alice's site and Bob's site), and that a node may have interfaces attached to different zones of the same scope (e.g., having one interface attached to Alice's site and another to Bob's site), a node requires an internal means of identifying to which zone a non-global address belongs. This is accomplished by assigning, within the node, a distinct "zone index" to each zone of the same scope to which that node is attached, and allowing all internal uses of an address to be qualified by a zone index.

The assignment of zone indices is illustrated in the example in the figure below:

organization). These site-local zones are identified by the site indices 1 and 2.

Note that each attached zone of the same scope must be assigned a different index value, whereas attached zones of different scopes can re-use the same index.

The zone indices are strictly local to the node. For example, the node on the other end of the point-to-point link may well be using entirely different interface, link, and site index values for that link.

An implementation should also support the concept of a "default" zone for each scope. It is convenient to reserve the index value zero, at each scope, to mean "use the default zone". This default index can also be used as the zone qualifier for an address for which the node is attached to only one zone, e.g., when using global addresses.

There is at present no way for a node to automatically determine which of its interfaces belong to the same zones, e.g., the same link or the same site. In the future, protocols may be developed to determine that information. In the absence of such protocols, an implementation must provide a means for manual assignment and/or reassignment of zone indices. Furthermore, to avoid the need to perform manual configuration in most cases, an implementation should, by default, initially assign zone indices as follows, and only as follows:

- o A unique interface index for each interface
- o A unique link index for each interface
- o A unique subnet (multicast "scop" value 3) index for each interface

Then, manual configuration would be necessary only for the less common cases of nodes with multiple interfaces to a single link or a single subnet, interfaces to different sites, or interfaces to zones of different (multicast-only) scopes.

Thus, the default zone index assignments for the example node from Figure 1 would be as illustrated in Figure 2, below. Manual

configuration would then be required to, for example, assign the same link index to the two Ethernet interfaces as shown in Figure 1.

zone as long as it meets the requirement that the index value of each attached zone of the same scope be unique within the node. Similarly, an implementation may choose an index value other than zero to represent the default zone. Implementations choosing to follow the recommended basic API [[BASICAPI](#)] will want to restrict their index values to those that can be represented by the `sin6_scope_id` field of a `sockaddr_in6`.

[7.](#) Sending Packets

When an upper-layer protocol sends a packet to a non-global destination address, it must have a means of identifying to the IPv6 layer the intended zone, for cases in which the node is attached to more than one zone of the destination address's scope.

Although identification of an outgoing interface is sufficient to identify an intended zone (because each interface is attached to no more than one zone of each scope), that is more specific than desired in many cases. For example, when sending to a site-local unicast address, from a node that has more than one interface to the intended site, the upper layer protocol may not care which of those interfaces is used for the transmission, but rather would prefer to leave that choice to the routing function in the IP layer. Thus, the upper-layer requires the ability to specify a zone index, rather than an interface identifier, when sending to a non-global, non-loopback destination address.

[8.](#) Receiving Packets

When an upper-layer protocol receives a packet containing a non-global source or destination address, the zone to which that address pertains can be determined from the arrival interface, because the arrival interface can be attached to only one zone of the same scope as the address under consideration. However, it is recommended that the IP layer convey to the upper layer the correct zone indices for the arriving source and destination addresses, in addition to the arrival interface identifier.

[9.](#) Forwarding

When a router receives a packet addressed to a node other than itself, it must take the zone of the destination and source addresses into account as follows:

- o The zone of the destination address is determined by the scope of the address and arrival interface of the packet. The next-hop interface is chosen by looking up the destination address in a (conceptual) routing table specific to that zone. That routing table is restricted to refer only to interfaces belonging to that zone.
- o After the next-hop interface is chosen, the zone of the source address is considered. As with the destination address, the zone of the source address is determined by the scope of the address and arrival interface of the packet. If transmitting the packet on the chosen next-hop interface would cause the packet to leave the zone of the

source address, i.e., cross a zone boundary of the scope of the source address, then the packet is discarded and an ICMP Destination Unreachable message [[RFC 2463](#)] with Code 2 ("beyond scope of source address") is sent to the source of the packet.

Note that the above procedure applies for addresses of all scopes, including link-local. Thus, if a router receives a packet with a link-local destination address that is not one of the router's own link-local addresses on the arrival link, the router is expected to try to forward the packet to the destination on that link (subject to successful determination of the destination's link-layer address via the Neighbor Discovery protocol [[RFC 2461](#)]). The forwarded packet may be transmitted back out the arrival interface, or out any other interface attached to the same link.

A node that receives a packet addressed to itself and containing a Routing Header with more than zero Segments Left [[RFC 2460](#), [section 4.4](#)] swaps the original destination address with the next address in the Routing Header. Then the above forwarding rules are applied, using the new destination address where the zone of the new destination address should be determined by the scope of the previous destination address and the interface to which the previous address belongs (which is not necessarily equal to the incoming interface). An implementation MUST NOT examine additional addresses in the Routing header to determine whether they are crossing boundaries for their scopes. Thus, it is possible, though generally inadvisable, to use a Routing Header to convey a non-global address across its associated zone boundary.

As an example, the router in Figure 4 must exchange routing information on five interfaces. The information exchanged is as follows:

- o Interface 1
 - o All global prefixes
 - o All site prefixes learned from Interfaces 1, 2, and 3
- o Interface 2
 - o All global prefixes
 - o All site prefixes learned from Interfaces 1, 2, and 3
- o Interface 3
 - o All global prefixes
 - o All site prefixes learned from Interface 1, 2, and 3
- o Interface 4
 - o All global prefixes
 - o All site prefixes learned from Interface 4

- o Interface 5
 - o All global prefixes
 - o All site prefixes learned from Interfaces 5

By imposing route exchange rules, zone integrity is maintained by keeping all zone-specific routing information contained within the zone.

11. Mobility

A mobile node using [[MOBILE](#)] that moves outside its "home site"

should not expect to be able to send and receive packets as if it had remained in the zone. In particular, the mobile node MUST NOT try to have a tunnel back into its old zone for the purposes of attempting such communication. This also implies that the mobile node should choose global addresses as home address whenever possible. This restriction should apply whether the scope of the zone is link-local or site-local.

Since there is no standard way to provide an ability to tell whether a mobile node is in its home site and/or whether a correspondent node is in the same site as the mobile node, the mobile node should always use a global care-of address.

[12.](#) Textual Representation

As already mentioned, to specify an IPv6 non-global address without ambiguity, an intended scope zone should be specified as well. As a common notation to specify the scope zone, an implementation SHOULD support the following format.

```
<address>%<zone_id>
```

where

```
<address> is a literal IPv6 address,  
<zone_id> is a string to identify the scope type and zone of  
the address, and  
'%' is a delimiter character to distinguish between <address>  
and <zone_id>.
```

The following subsections describe detail definitions, concrete examples, and additional notes of the format.

12.1 Non-Global Addresses

The format is applied to all kinds of unicast and multicast addresses of non-global scope. Although the format is meaningless and should

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

12

Internet Draft

IPv6 Scoped Address Architecture

November 2001

not be used for global addresses, an implementation which handles addresses (e.g. name to address mapping functions) MAY allow users to use such a notation.

12.2 Zone Indices

In the textual representation, a zone index should be able to

identify a particular type of scope and a particular zone of the scope. Note that the representation contains the type of scope as well as the identifier within the particular scope type. This can be useful, for example, when the zone index without an address is used in a Management Information Base (MIB).

The actual representation of the zone indices, including how to specify the scope type, is implementation dependent. But one possible example would be to use an integer in which the upper-most 4 bits specifies the scope type just like the "scop" subfield of multicast addresses and the rest of the value specifies a particular zone of the scope. For instance, site zones in Figure 1 of [Section 6](#) can be represented as 32-bit integers 50000001 and 50000002 (in hexadecimal), which designate site1 and site2, respectively.

When a zone index is used with an address in the form of <address>%<zone_id>, the scope type of the address MUST be equal to the scope type of the zone index. An implementation SHOULD check the consistency when it interprets the format.

An implementation SHOULD support at least numerical indices as <zone_id>, which are non-negative decimal integers. Positive indices MUST uniquely specify a single zone of a single scope type. An implementation MAY use zero ("0") to specify a "default" zone as described in [Section 6](#) of this document. In this case, the corresponding scope type is the scope type of the <address> part.

Since the identifiers contain scope type values, the decimal integers may not be intuitive for operators. For example, if the upper-most 4 bits were used for the scope type values, "site1" would be represented as 1342177281 (which is 50000001 in hexadecimal).

In order to improve the readability, this document also defines aliases for the decimal representation. The alias is constructed by concatenating a string and a decimal number, where the string specifies the scope type and the number identifies a particular zone of the scope. Today there are five non-global scope types defined; interface-local, link-local, subnet-local, admin-local, and organization-local, and this document defines five scope type strings accordingly; "intf", "link", "sbnt", "admn", "site", and "orgn". Using this alias, a string "site1" can be used as well as "1342177281" in the example above. An alias string for the global scope is intentionally undefined, since there is no ambiguity about

the global scope and it is practically expected zone indices are omitted for global addresses.

An implementation MAY support other kinds of non-null strings as `<zone_id>`. However, the strings must not conflict with the delimiter character or with the aliases. The precise format and semantics of such additional strings is implementation dependent.

One possible candidate of such strings would be interface names, since interfaces uniquely disambiguate any type of scopes. In particular, interface names can be used as "default identifiers" for interfaces, links, and subnets, because there is, by default, a one-to-one mapping between interfaces and each of those scopes as described in [Section 6](#).

An implementation could also use interface names as `<zone_id>` for larger scopes than subnets, but there might be some confusion in such use. For example, when more than one interface belongs to a same site, a user would be confused about which interface should be used. Also, a mapping function from an address to a name would encounter a same kind of problem, when it prints an address with an interface name as a zone index. This document does not specify how these cases should be treated and leaves it implementation dependent.

It cannot be assumed that a same index is common to all nodes in a zone (see [Section 6](#)). Hence, the format MUST be used only within a node and MUST NOT be sent on a wire unless every node that interprets the format agrees with the semantics.

12.3 Examples

Here are examples. The following addresses

```
fe80::1234 (on the 1st link of the node)
fec0::5678 (on the 2nd site of the node)
ff02::9abc (on the 5th link of the node)
ff08::def0 (on the 10th organization of the node)
```

would be represented as follows:

```
fe80::1234%536870913 or fe80::1234%link1
fec0::5678%1342177282 or fec0::5678%site2
ff02::9abc%536870917 or ff02::9abc%link5
ff08::def0%2147483658 or ff08::def0%orgn10
```

(Here we assume 32-bit integers as zone indices with the upper-most 4 bits specifying the scope type.)

If we use interface names as `<zone_id>`, those addresses could also be

represented as follows:

```
fe80::1234%ne0
fec0::5678%ether2
ff02::9abc%pvc1.3
ff08::def0%interface10
```

where the interface "ne0" belongs to 1st link, "ether2" belongs to 2nd site, and so on.

12.4 Usage Examples

Applications that are supposed to be used in end hosts like telnet, ftp, and ssh, may not explicitly support the notion of address scope, especially of link-local addresses. However, an expert user (e.g. a network administrator) sometimes has to give even link-local addresses to such applications.

Here is a concrete example. Consider a multi-linked router, called "R1", that has at least two point-to-point interfaces (links). Each of the interfaces is connected to another router, called "R2" and "R3", respectively. Also assume that the point-to-point interfaces are "unnumbered", that is, they have link-local addresses only.

Now suppose that the routing system on R2 hangs up and has to be reinvoked. In this situation, we may not be able to use a global address of R2, because this is a routing trouble and we cannot expect that we have enough routes for global reachability to R2.

Hence, we have to login R1 first, and then try to login R2 using link-local addresses. In such a case, we have to give the link-local address of R2 to, for example, telnet. Here we assume the address is fe80::2.

Note that we cannot just type like

```
% telnet fe80::2
```

here, since R1 has more than one link and hence the telnet command cannot detect which link it should try to connect. Instead, we should type the link-local address with the link index as follows:

```
% telnet fe80::2%link3
```

where "link3" after the delimiter character '%' is the link index of the point-to-point link.

Another example is an EBGp peering. When two IPv6 ISPs establish an EBGp peering, using a particular ISP's global addresses for the peer would be unfair, and using their link-local addresses would be better in a neutral IX. In such a case, link-local addresses should be

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

15

Internet Draft

IPv6 Scoped Address Architecture

November 2001

specified in a router's configuration file and the link for the addresses should be disambiguated, since a router usually connects to multiple links.

12.5 Related API

The "Basic Socket API" [[BASICAPI](#)] defines how the format for non-global addresses should be treated in library functions that translate a nodename to an address, or vice versa.

12.6 Omitting Zone Indices

The format defined in this document does not intend to invalidate the original format for non-global addresses, that is, the format without the zone index portion. An implementation SHOULD rather provide a user with a "default" zone of each scope and allow the user to omit zone indices in textual representations.

Also, when an implementation can assume that there is no ambiguity of any type of scopes on a node, it MAY even omit the whole functionality to handle the format. An end host with a single interface would be an example of such a case.

12.7 Combinations of Delimiter Characters

There are other kinds of delimiter characters defined for IPv6 addresses. In this subsection, we describe how they should be combined with the format for non-global addresses.

The IPv6 addressing architecture [[ADDRARCH](#)] also defines the syntax of IPv6 prefixes. If the address portion of a prefix is non-global and its scope zone should be disambiguated, the address portion SHOULD be in the format. For example, the prefix fec0:0:0:1::/64 on the 2nd site should be represented as follows:

fec0:0:0:1::%site2/64

In this combination, it is important to place the zone index portion before the prefix length, when we consider parsing the format by a name-to-address library function [[BASICAPI](#)]. That is, we can first separate the address with the zone index from the prefix length, and just pass the former to the library function.

The preferred format for literal IPv6 addresses in URL's are also defined [[RFC 2732](#)]. When a user types the preferred format for an IPv6 non-global address whose zone should be explicitly specified, the user could use the format for the non-global address combined with the preferred format.

However, the typed URL is often sent on a wire, and it would cause confusion if an application did not strip the <zone_id> portion

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

16

Internet Draft

IPv6 Scoped Address Architecture

November 2001

before sending. Also, the format for non-global addresses might conflict with the URI syntax [[RFC 2396](#)], since the syntax defines the delimiter character ('%') as the escape character.

Hence, this document does not specify how the format for non-global addresses should be combined with the preferred format for literal IPv6 addresses. As for the conflict issue with the URI format, it would be better to wait until the relationship between the preferred format and the URI syntax is clarified. In fact, the preferred format for IPv6 literal addresses itself has same kind of conflict. In any case, it is recommended to use an FQDN instead of a literal IPv6 address in a URL, whenever an FQDN is available.

[13. Related Documents](#)

The following documents have aspects related to IPv6 address scope, but are not cited elsewhere in this document:

- o Default Address Selection for IPv6, [draft-ietf-ipngwg-default-addr-select-06.txt](#)

[14. Security Considerations](#)

The routing section of this document specifies a set of guidelines that allow routers to prevent zone-specific information from leaking

out of each site. If site boundary routers allow site routing information to be forwarded outside of the site, the integrity of the site could be compromised.

Since the use of the textual representation of non-global addresses is restricted within a single node, it does not create a security vulnerability from outside the node. However, a malicious node might send a packet that contains a textual IPv6 non-global address with a zone index, intending to deceive the receiving node about the zone of the non-global address. Thus, an implementation should be careful when it receives packets that contain textual non-global addresses as data.

15. References

[RFC 2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), [BCP14](#), March 1999.

[ADDRARCH] Hinden, R., and Deering, S., "IP Version 6 Addressing Architecture", Internet Draft, [draft-ietf-ipngwg-addr-arch-v3-04.txt](#), February 2001.

[RFC 2460] Deering, S., and Hinden, R., "Internet Protocol Version

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

17

Internet Draft

IPv6 Scoped Address Architecture

November 2001

6 (IPv6) Specification", [RFC 2460](#), December 1998.

[RFC 2473] Conta, A., and Deering, S., "Generic Packet Tunneling in IPv6 Specification", [RFC 2473](#), December 1998.

[RFC 2463] Conta, A., and Deering, S., "Internet Control Message Protocol ([RFC 2463](#)) for Internet Protocol Version 6 (IPv6)", [RFC 2463](#), December 1998.

[RFC 2461] Narten, T., Nordmark, E., and Simpson, W., "Neighbor Discovery for IP Version 6 (IPv6)", [RFC 2461](#), December 1998.

[MOBILE] Johnson, D.B., and Perkins, C., "Mobility Support in IPv6", Internet Draft, [draft-ietf-mobileip-ipv6-14.txt](#), July 2001.

[BASICAPI] Gilligan, R. E., Thomson, S., Bound, J., Stevens, W., "Basic Socket Interface Extensions for IPv6", Internet Draft, [draft-ietf-ipngwg-rfc2553bis-03.txt](#), February 2001.

[RFC 2732] Hinden, R., Carpenter, B., Masinter, L., "Preferred Format for Literal IPv6 Addresses in URL's", [RFC 2732](#), December 1999.

[RFC 2396] T. Berners-Lee, R. Fielding, and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", [RFC 2396](#), August 1998.

Acknowledgements

Authors' Addresses

Stephen E. Deering
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA

Phone: +1-408-527-8213
Fax: +1-408-527-8213
Email: deering@cisco.com

Brian Haberman
No Affiliation

Phone: +1-919-949-4828
Email: haberman@innovationslab.net

Deering, Haberman, Jinmei, Nordmark, Onoe, Zill

18

Internet Draft

IPv6 Scoped Address Architecture

November 2001

Tatuya JINMEI
Corporate Research & Development Center, Toshiba Corporation
1 Komukai Toshiba-cho, Kawasaki-shi
Kanagawa 212-8582, JAPAN

Phone: +81-44-549-2230
Fax: +81-44-520-1841
Email: jinmei@isl.rdc.toshiba.co.jp

Erik Nordmark

Sun Microsystems Laboratories, Europe
29 Chemin du Vieux Chene
38240 Meylan, France

Phone: +33 (0)4 76 18 88 03
Fax: +33 (0)4 76 18 88 88
Email: Erik.Nordmark@sun.com

Atsushi Onoe
IT Development Division, NSC, Sony Corporation
6-7-35 Kitashinagawa, Shinagawa-ku
Tokyo 141-0001, JAPAN

Phone: +81-3-5475-8491
Fax: +81-3-5475-8977
Email: onoe@sm.sony.co.jp

Brian D. Zill
Microsoft Research
One Microsoft Way
Redmond, WA 98052-6399
USA

Phone: +1-425-703-3568
Fax: +1-425-936-7329
Email: bzill@microsoft.com