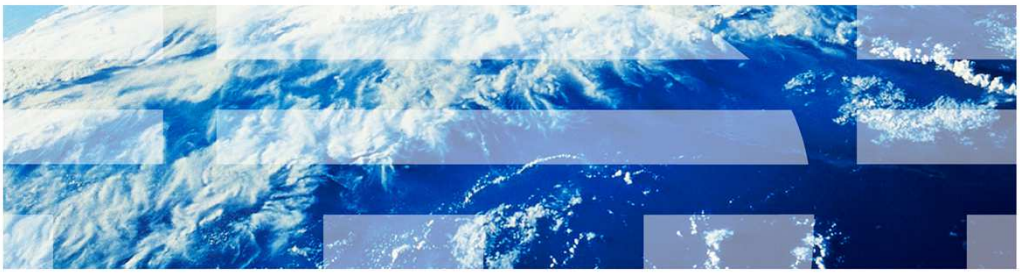
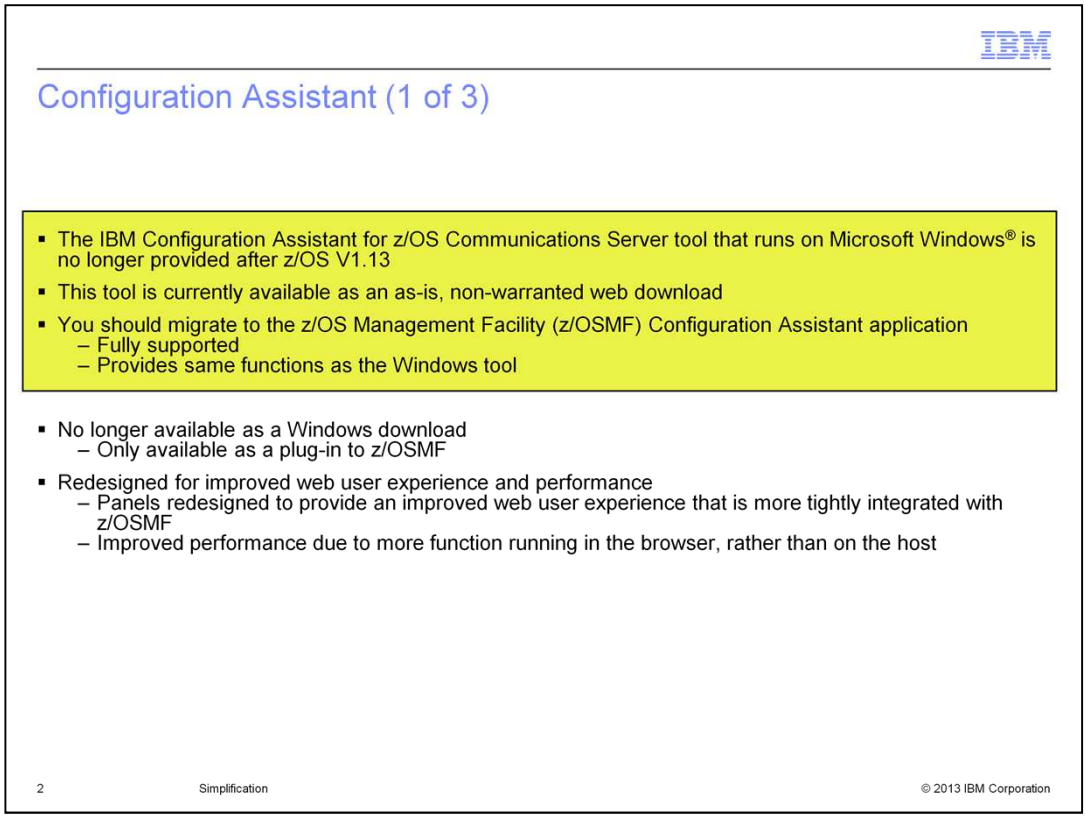
z/OS V2R1 Communications Server

Simplification

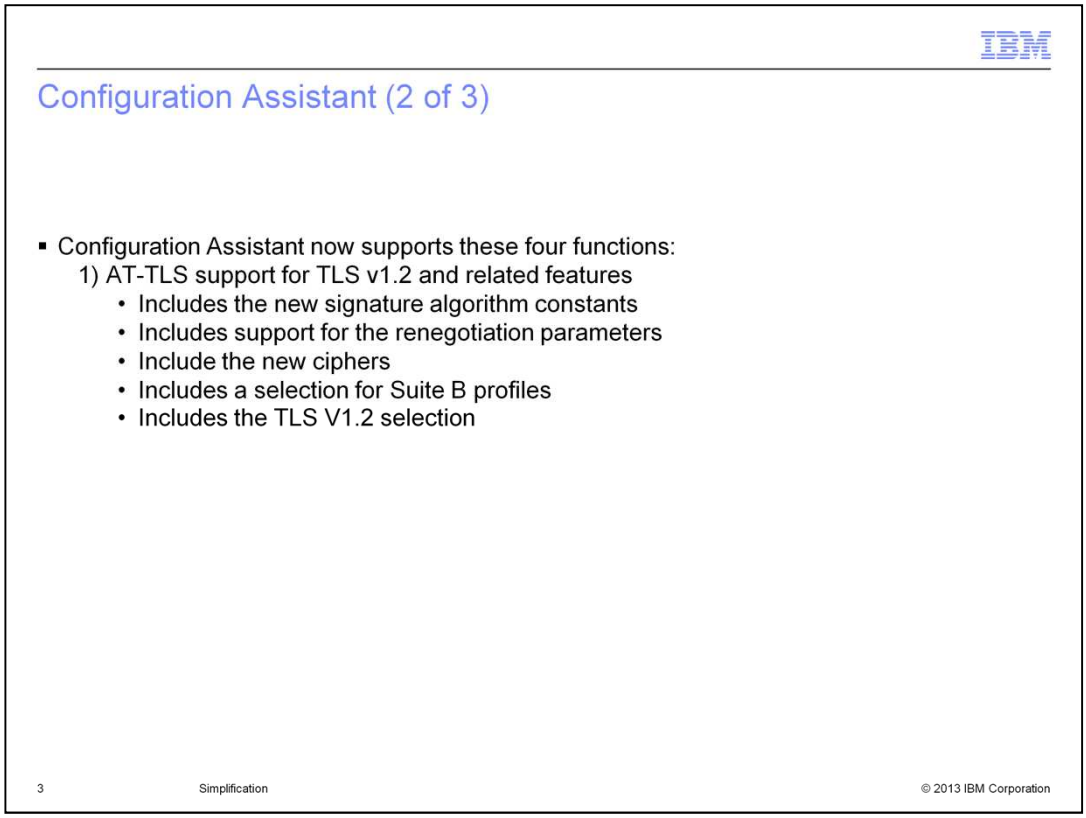


This presentation provides an overview of the z/OS® V2R1 Communications Server enhancements for simplification.



Configuration Assistant for z/OS Communications Server is a GUI for configuring z/OS Communications Server policy-based networking technologies. It is a z/OSMF application. Customers using the Windows version of the IBM Configuration Assistant for z/OS Communications Server should migrate to the z/OSMF Configuration Assistant application.

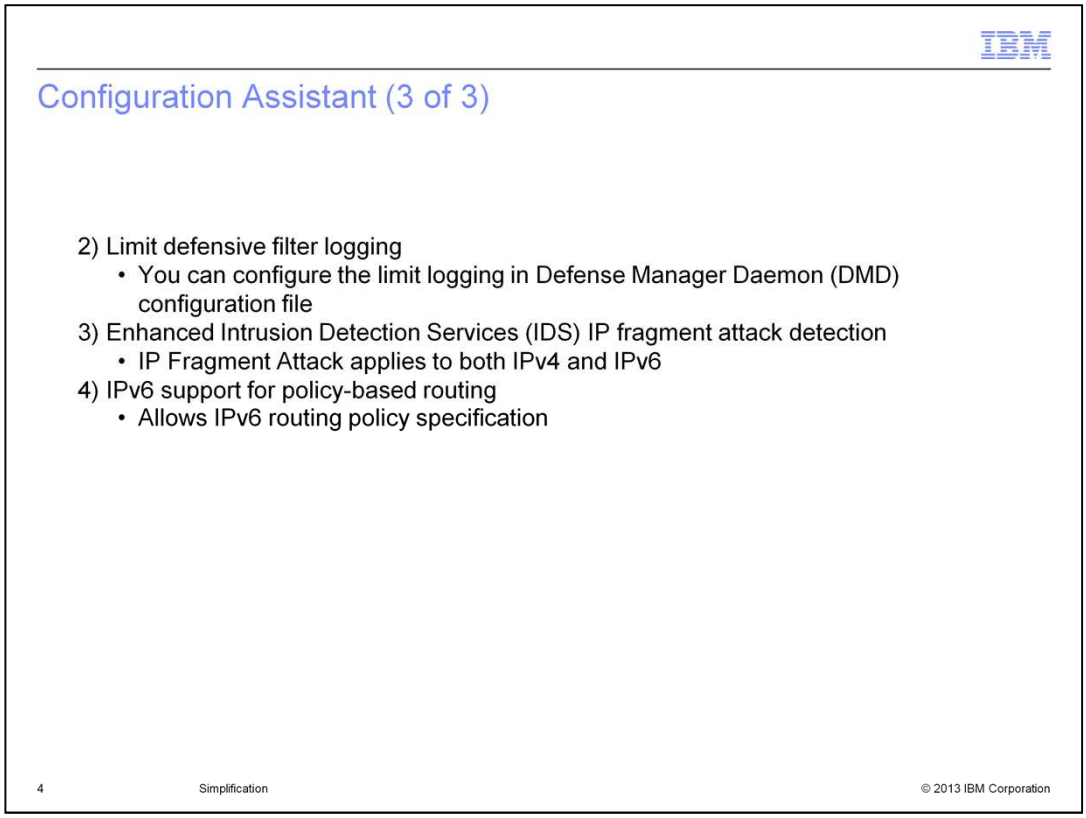
In V2R1, Configuration Assistant has been redesigned for an improved web user experience and improved performance.



In V2R1, the IBM Configuration Assistant for z/OS Communications Server plug-in that runs within z/OSMF now supports several z/OS V2R1 Communications Server functions.

First, new Application Transparent Transport Layer Security parameters are supported. The new signature algorithm constants, the renegotiation parameters, the new ciphers, Suite B profiles, and the new Transport Layer Security V1.2 option can all be selected from Configuration Assistant.

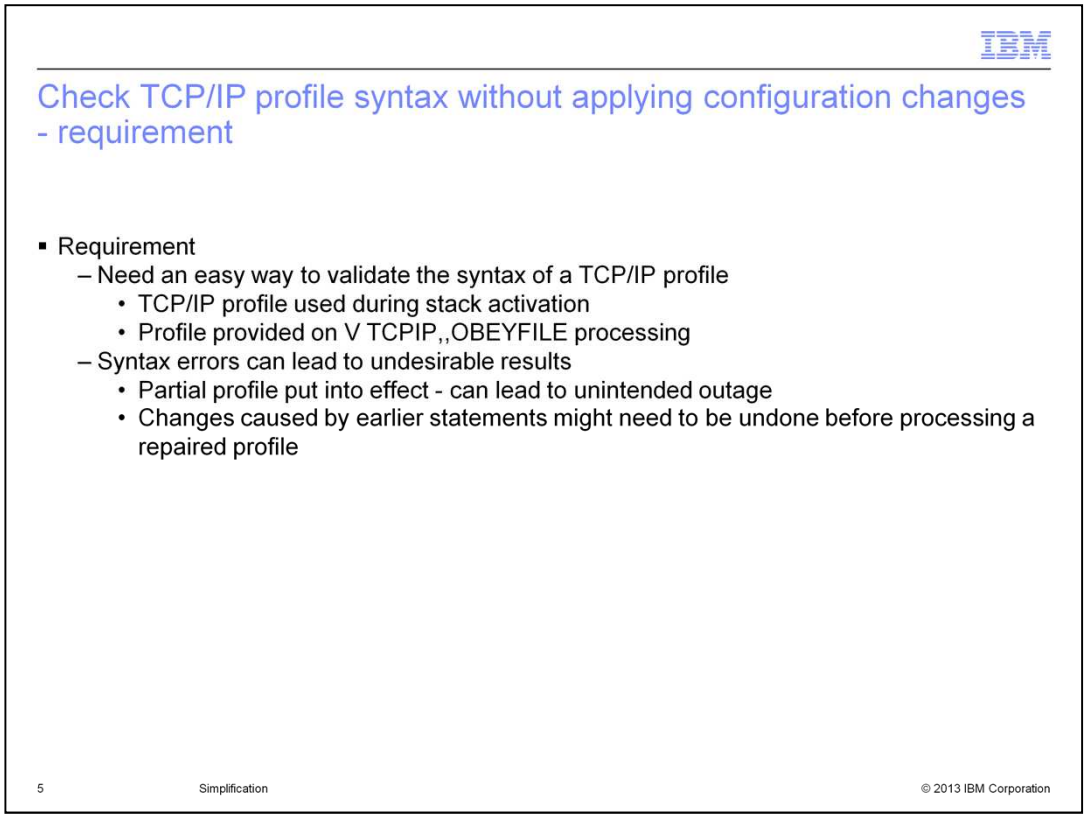
Configuration Assistant can configure multiple z/OS releases. However, the new Application Transparent Transport Layer Security parameters are not available in releases V1R12 or V1R13. Configuration Assistant will warn you about differences between releases.



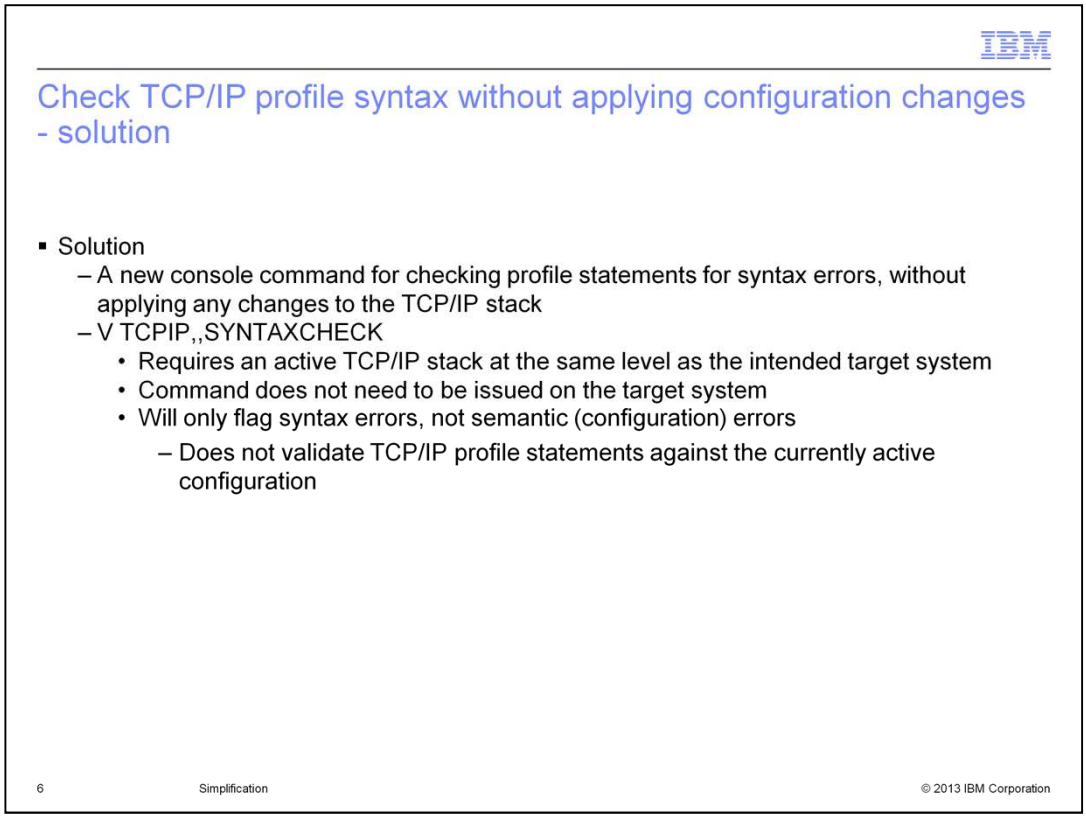
Second, you can use Configuration Assistant to create and update your Defense Manager Daemon configuration. In support of the limit defensive filter logging function, Configuration Assistant provides an enhanced panel that you use to enter Defense Manager Daemon configuration information for a TCP/IP stack for which defensive filtering is being done. You can check the “Limit logging” box and then select a limit from 1 – 9999. This provides a default log limit for defensive filters that are added to the stack. A filter that is added without specifying log limit on the *ipsec -F add* command inherits this default log limit. If you do not want to limit the number of messages written to syslogd for packets matching a defensive filter, leave the box unchecked. The default is then unlimited logging.

Third, in Configuration Assistant, within the Intrusion Detection Services perspective and Intrusion Detection Services requirement maps, you can enable the attack type of IP Fragment Attack to enable the IP fragment attack protection. IP Fragment Attack now includes both IPv4 and IPv6 fragments.

And finally, in support of IPv6 policy-based routing, Configuration Assistant panels that are used to configure a source IP address value and a destination IP address value for a routing rule now allow an IPv6 address specification. Updates were also made to the panel that is used to configure a static route for a policy-based route table. You can now indicate whether a route is an IPv4 route or an IPv6 route. You can specify IPv6 values in both the destination address field and the first hop address field. You can also specify the name of an IPv6 interface and an IPv6 first hop value in a dynamic routing parameter.

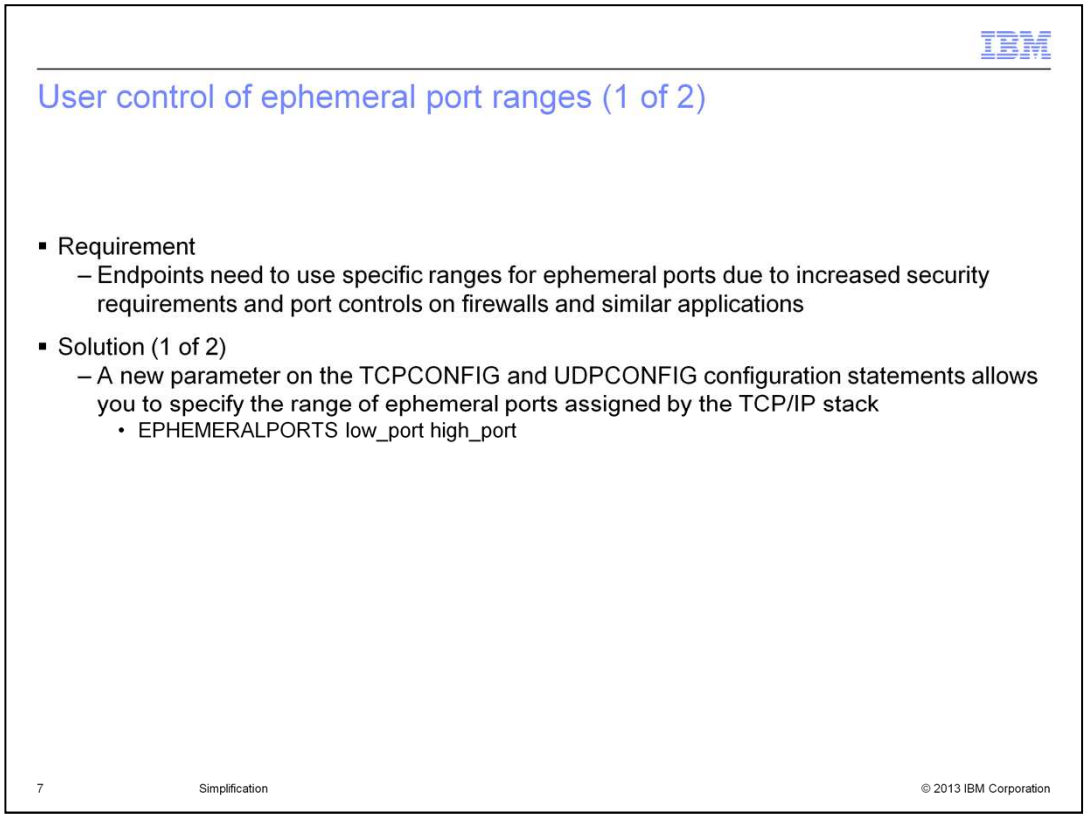


A TCP/IP profile is the data set or collection of data sets that contains statements to configure the TCP/IP stack. The stack reads the profile and processes the configuration statements when you start TCP/IP. The stack also reads and processes a profile when you issue the VARY TCPIP,,OBEYFILE,profile command to change the configuration. The command argument *profile* is the profile you want to activate. The profile parser performs a single pass of the TCP/IP profile. That is, the parser reads a statement, parses it and saves the information before going on to the next statement. When the profile parser finds a statement with syntax errors, it either discards the statement or it overrides the values coded on the statement with default values. When the information is passed to the protocol stack, the protocol stack might detect and report configuration errors, such as conflicts with the active configuration. Not all profile errors are syntax errors.



z/OS V2R1 Communications Server offers a new console command that you can use to check a profile for statement syntax errors, without applying any changes to the TCP/IP stack configuration. The command is called VARY TCPIP,,SYNTAXCHECK. This new console command will report errors using the same messages that initial profile processing and the VARY TCPIP,,OBEYFILE processing use.

The VARY TCPIP,,SYNTAXCHECK command supports MVS system symbols. If your profile statements use MVS system symbols, you should issue the command on the MVS system where you plan to use the profile, in order to get consistent resolution of the MVS system symbols. If your profile statements are coded without MVS system symbols, you can check the profile syntax using any TCP/IP stack that supports the VARY TCPIP,,SYNTAXCHECK command and the statements you coded in your profile. Communications Server does frequently add new configuration statements and sometimes change the rules for old configuration statements. A good practice is to always check profiles with a TCP/IP stack of the same z/OS release level as the TCP/IP stack that you will use to activate the profile. That way, the error messages that TCP/IP displays when processing a profile with the VARY TCPIP,,SYNTAXCHECK command are consistent with the messages it displays when activating a profile.



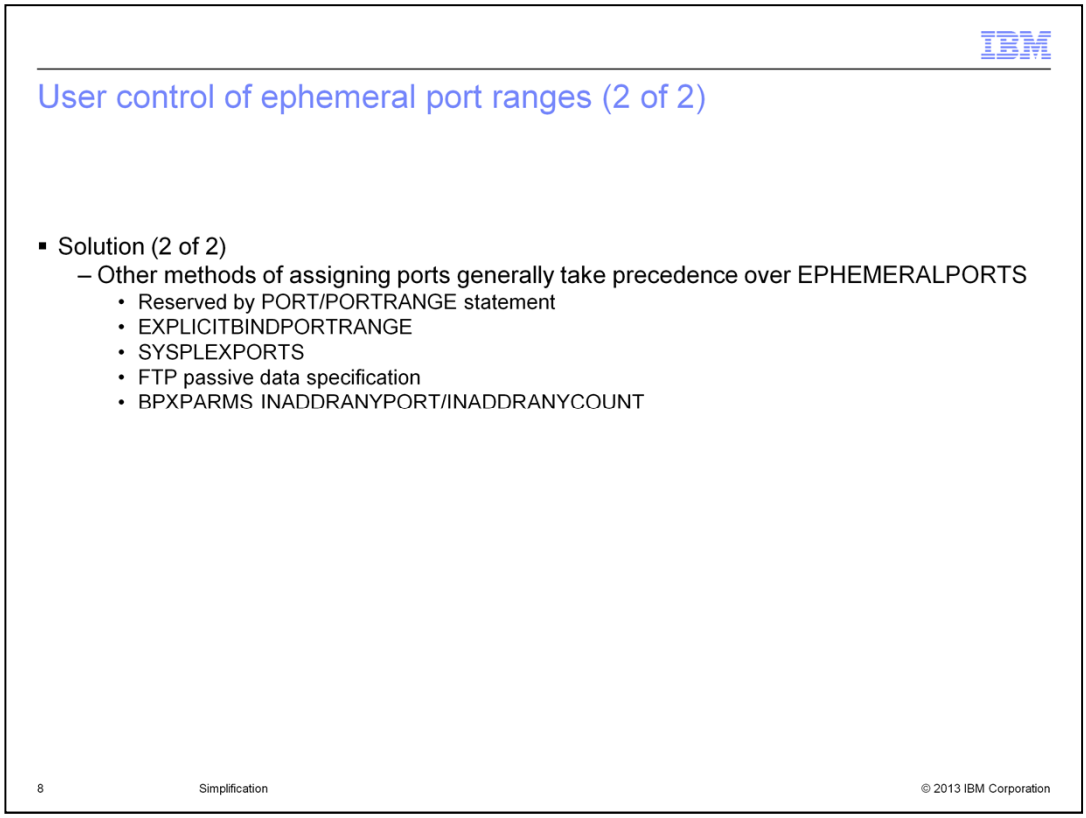
Typically, ephemeral ports are ports that the TCP/IP stack assigns to a client when the client issues a connect() socket call and the port number is not yet known. These ports are assigned to the client only for the duration of the connection. When the connection ends, TCP/IP is free to reuse the port number for a different connection or a different client.

In earlier releases, ephemeral ports are port numbers between 1024 and 65535. Some of those ports can be reserved for specific users using PORT and PORTRANGE statements in the TCP/IP profile data set.

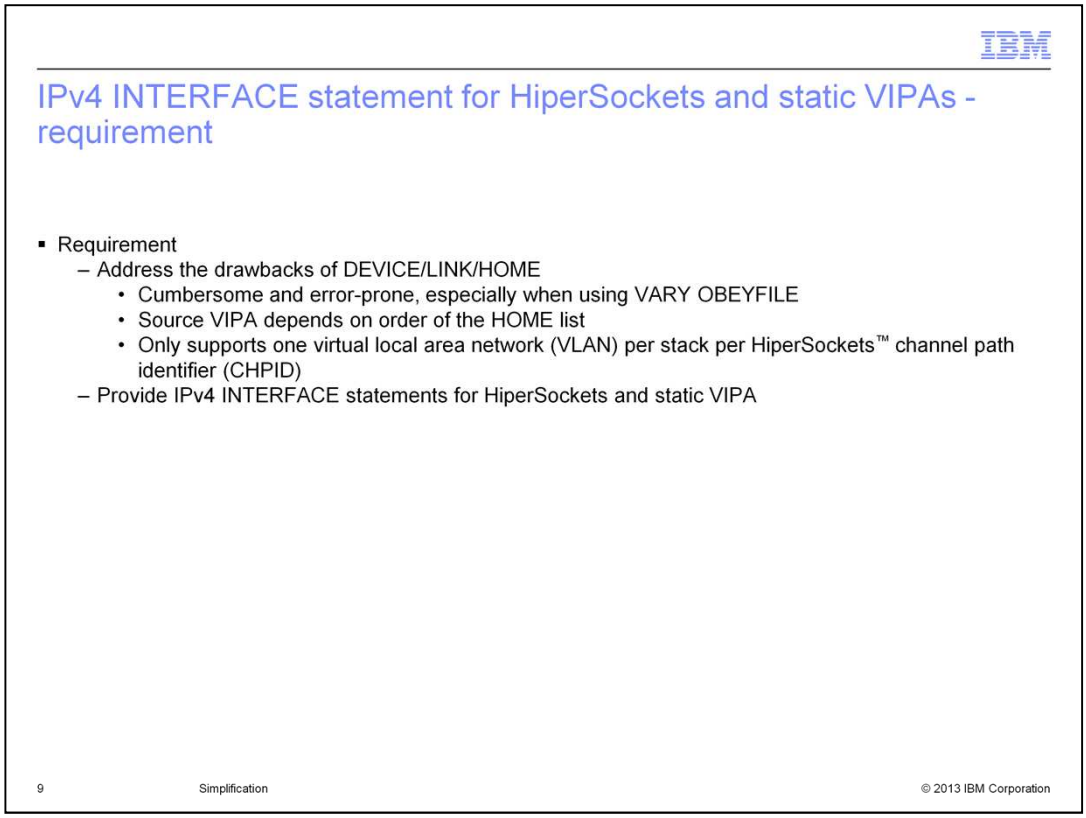
EXPLICITBINDPORTRANGE limits the ports assigned by the stack, but only when an application binds explicitly to the IPv4 inaddr_any address or to the IPv6 unspecified address (in6addr_any), and port 0.

Increased security requirements necessitate configuring firewalls to limit the range of acceptable ports.

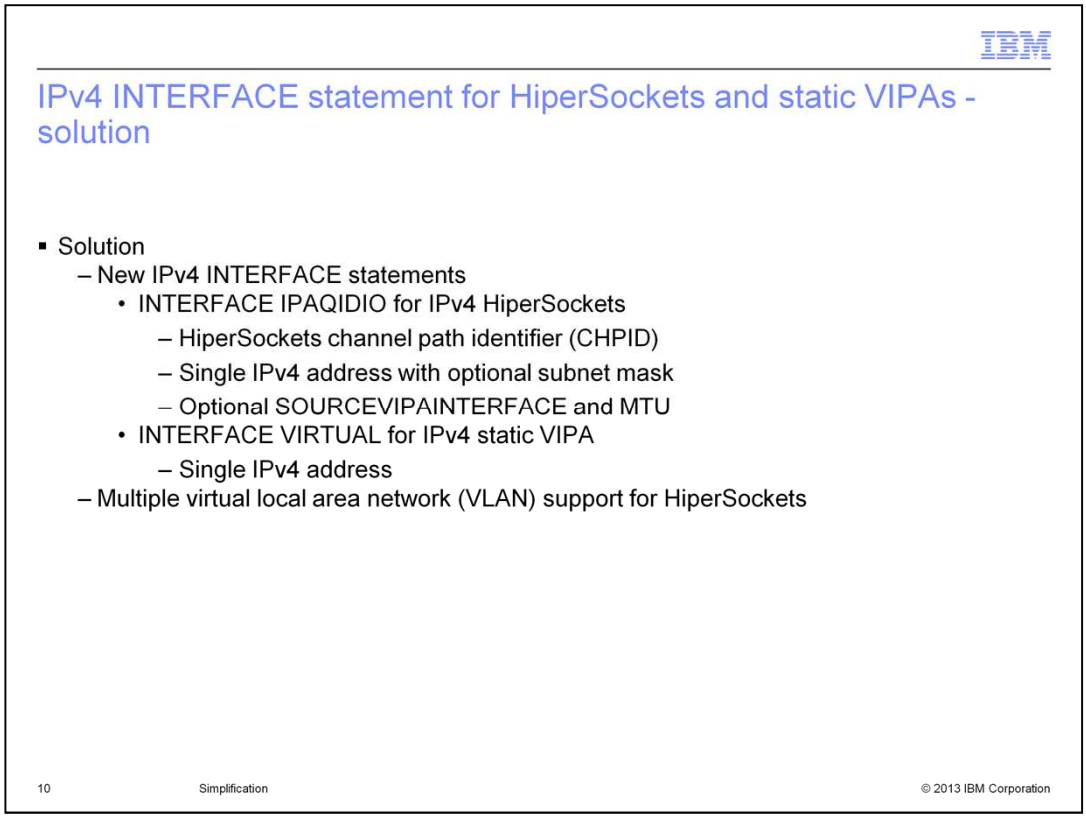
To satisfy this need, a new parameter, called EPHEMERALPORTS, on the TCPCONFIG and UDPCONFIG statements allows you to specify the low and high ephemeral ports that the stack will give out. Separate definitions for TCPCONFIG and UDPCONFIG statements allow you to specify different ranges per protocol. The default values for both protocols are 1024 and 65535. You must specify both the low and high port values.



Other methods of assigning or reserving ports can have interactions with EPHEMERALPORTS. In general, the other methods take precedence over EPHEMERALPORTS. The ports available for ephemeral port assignment are generally those in the range specified by EPHEMERALPORTS that are left after other methods of assigning ports have been applied.



In previous releases, DEVICE/LINK/HOME definitions are awkward and error-prone, especially when making profile changes using VARY OBEYFILE. For example, if you make any changes to the HOME list, you must specify a complete replacement for the HOME statement. Also, if you use source VIPA, the VIPA that gets associated with a DEVICE/LINK definition depends on the order of IP addresses on the HOME statement. Furthermore, with the DEVICE/LINK statement, a stack can only use a single virtual local area network for a HiperSockets channel path identifier and must use the same virtual local area network identifier for IPv4 and IPv6.

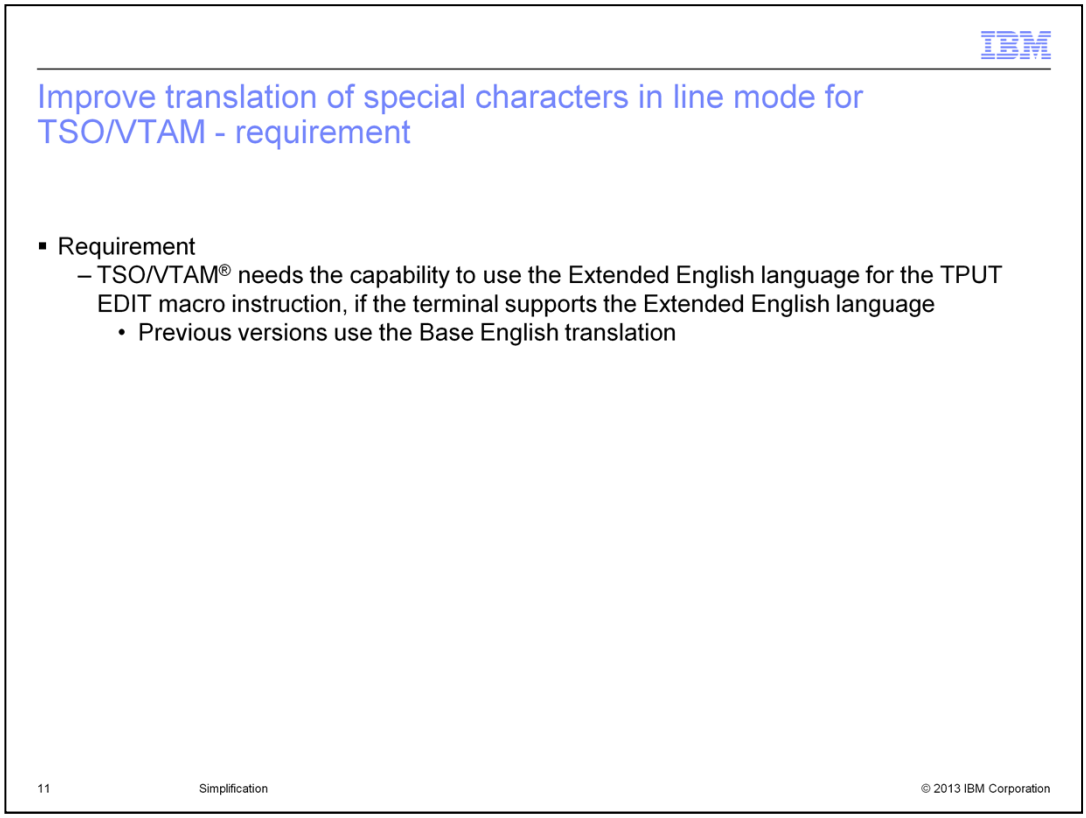


In V2R1, the new IPv4 INTERFACE statement for HiperSockets is similar to the existing IPv6 statement. The CHPID parameter specifies the HiperSockets channel path identifier. This new statement also has many characteristics in common with the existing IPv4 INTERFACE statement for OSA-Express Queued Direct I/O. Specifically, it requires a single IPv4 address, with an optional subnet mask, and has optional parameters for specifying the source VIPA and MTU for the interface.

The new IPv4 INTERFACE statement for static VIPA is similar to the existing IPv6 INTERFACE statement. It requires a single IPv4 address.

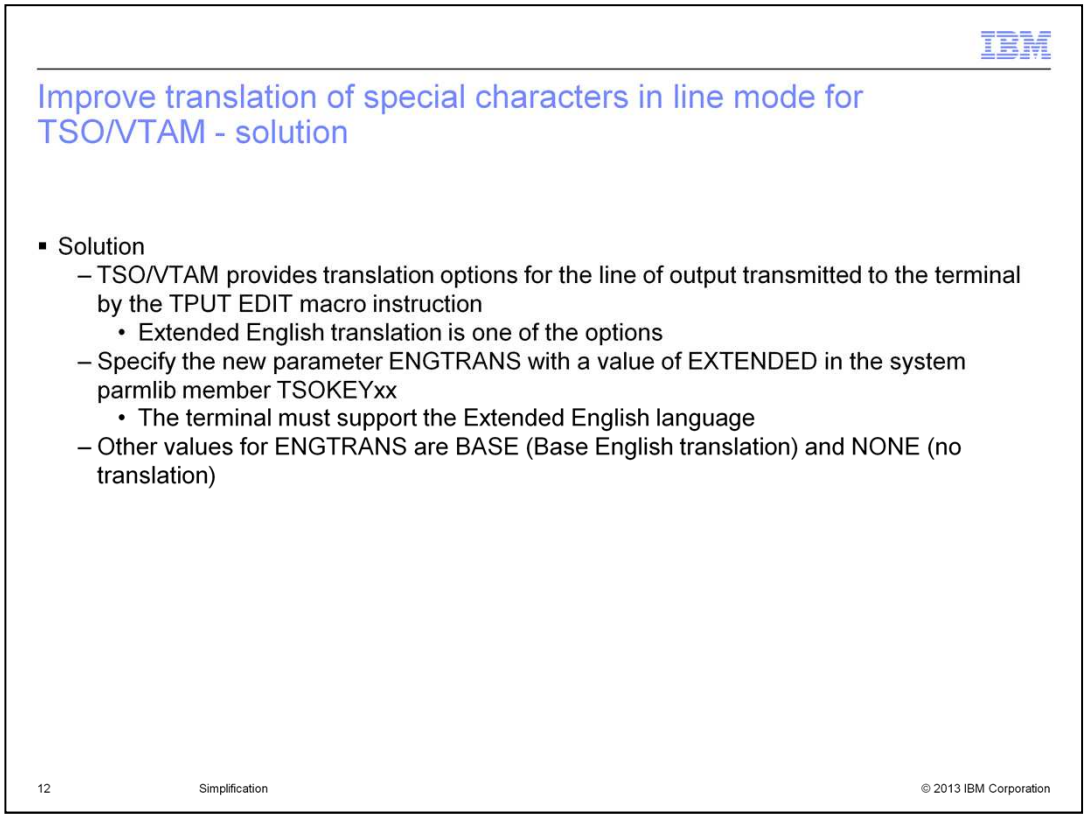
In z/OS V2R1, you can access multiple virtual local area networks for the same HiperSockets channel path identifier by configuring multiple interfaces for that channel path identifier with unique virtual local area network identifiers. Similar to the OSA-Express Queued Direct I/O support for multiple virtual local area networks, you must use the INTERFACE statement for each virtual local area network and, for IPv4, must configure a unique subnet for each of these definitions.

A z/OS image can have up to eight IPv4 interfaces and up to eight IPv6 interfaces for a single HiperSockets channel path identifier. If you have multiple stacks on the same z/OS image, the total number of virtual local area networks for these stacks cannot exceed these limits.

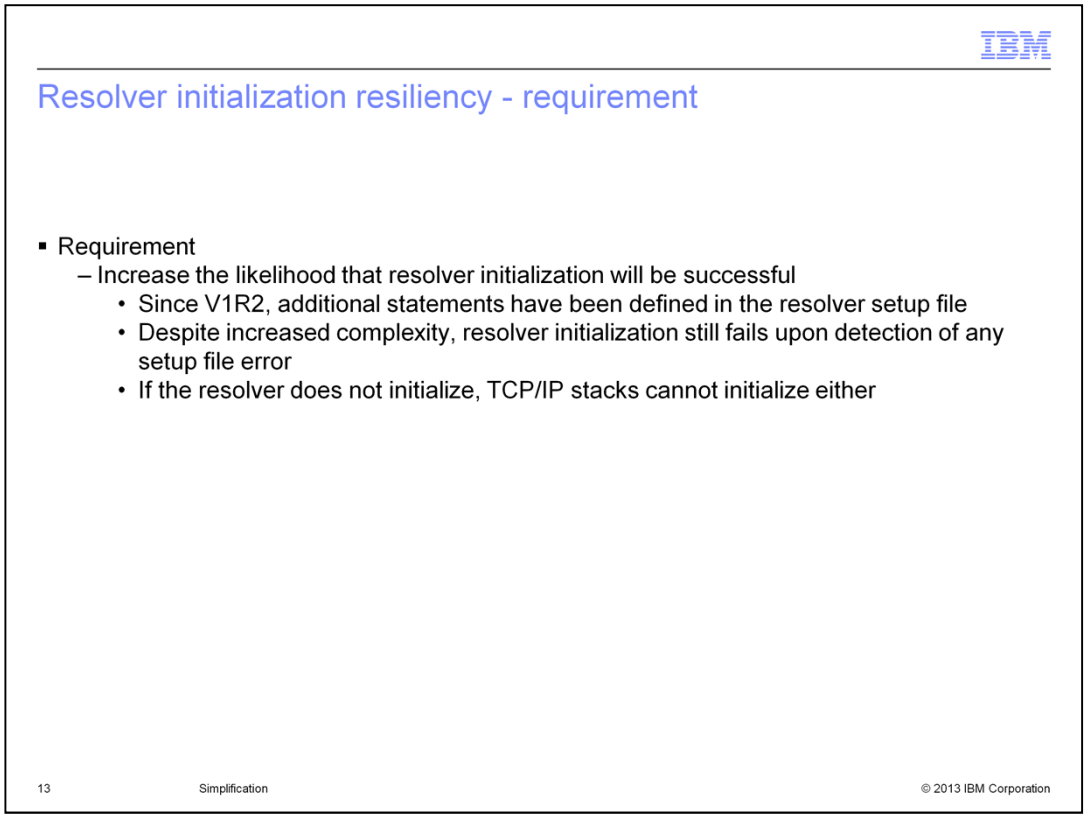


In prior releases, when a time sharing option user logs on to the terminal, TSO/VTAM queries the device to learn the alternate screen size. When the query information is returned, it also contains the language and character set CGCSGID supported by the device. When the Extended English character set CGCSGID X'02B90025' is returned, TSO/VTAM uses the Base English translation table to validate characters for the TPUT macro instruction with the EDIT operand specified.

Using the Base English translation for the TPUT macro instruction with the EDIT operand when the terminal supports the Extended English language causes problems for some people. For example, TSO/VTAM translates characters like - {},[] and \ to colons. These problems would be avoided if TSO/VTAM used the Extended English translation table, as described in the D/T3174 Character Set Reference manual.

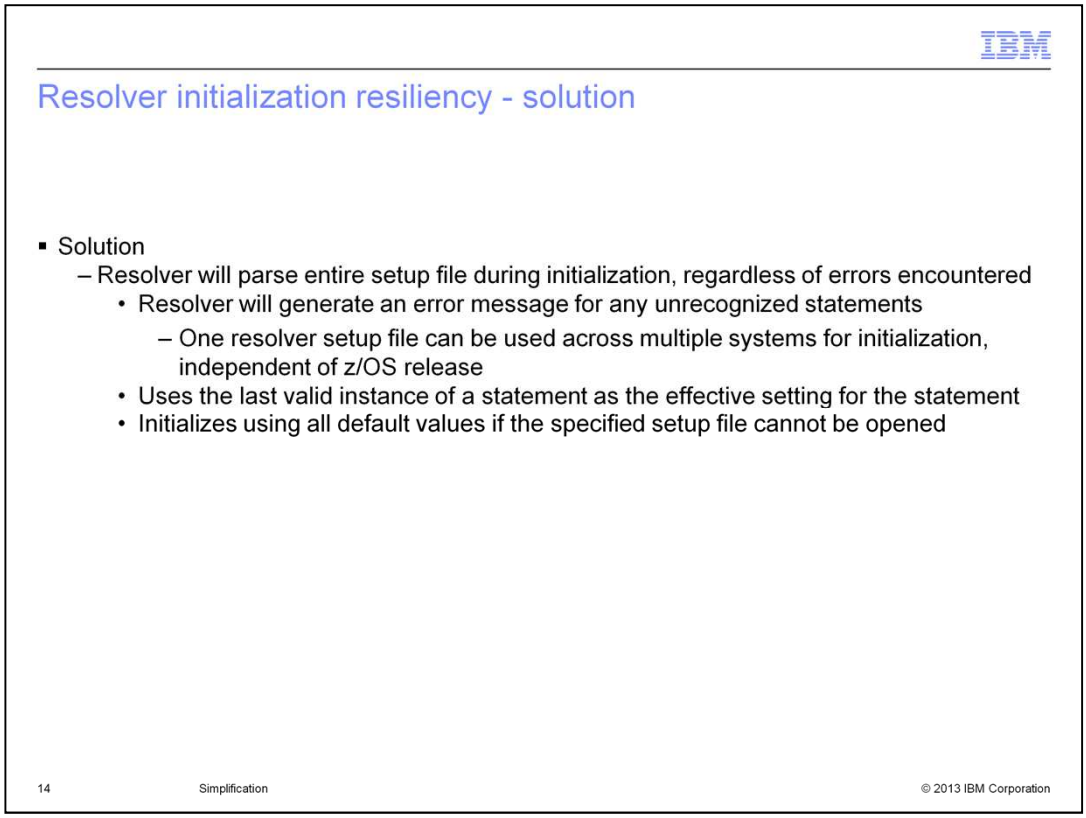


Starting in z/OS V2R1, TSO/VTAM offers three options for translating a line of output to a terminal/emulator that supports the Extended English character set. This support is available when the TPUT macro instruction with the EDIT operand is used. The translation options are: Base English translation, no translation, and Extended English translation. The EXTENDED option indicates the Extended English translation, as described in D/T3174 Character Set Reference manual.

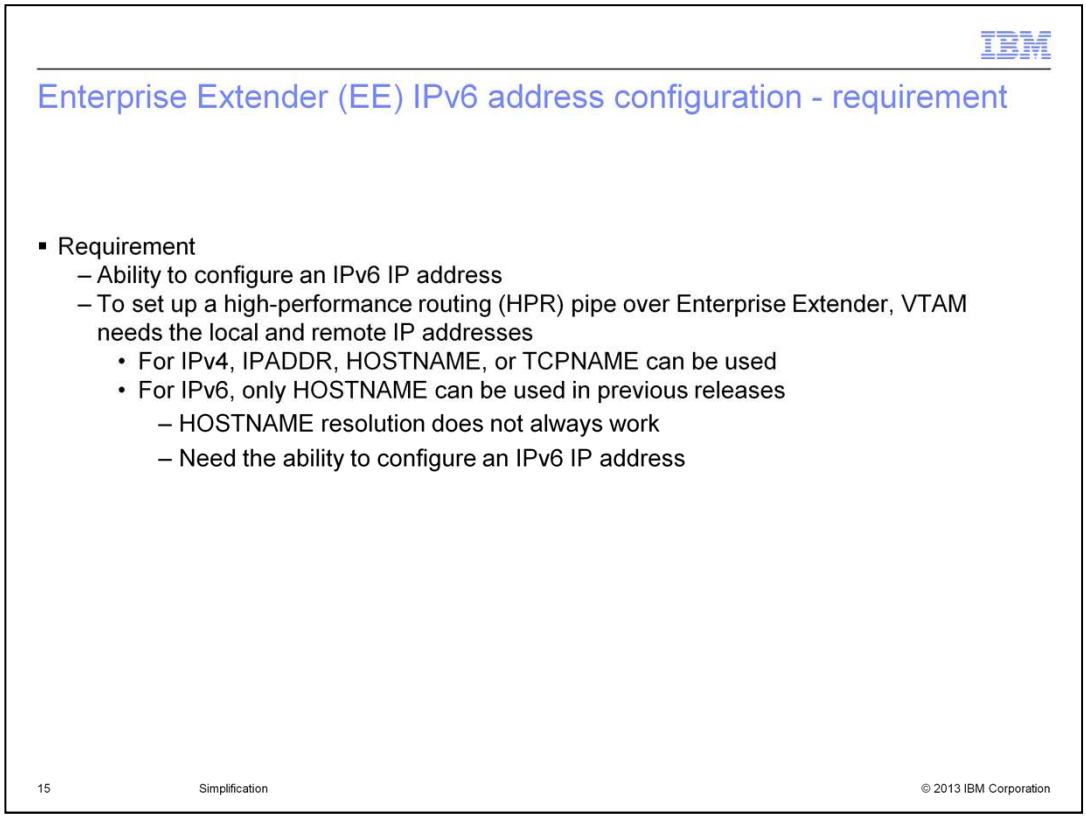


The resolver was introduced in z/OS V1R2. It was the result of merging several different resolvers into a single resolver to be used across the product. When the converged resolver was created, there were only two resolver configuration/setup statements: GLOBALTCPIPDATA and DEFAULTTCPIPDATA. You were not required to code either statement or create a resolver setup file. For that reason, any error encountered while parsing and processing these two statements causes resolver initialization to terminate with an error. In addition, any unrecognized statement also causes the resolver initialization to terminate with an error.

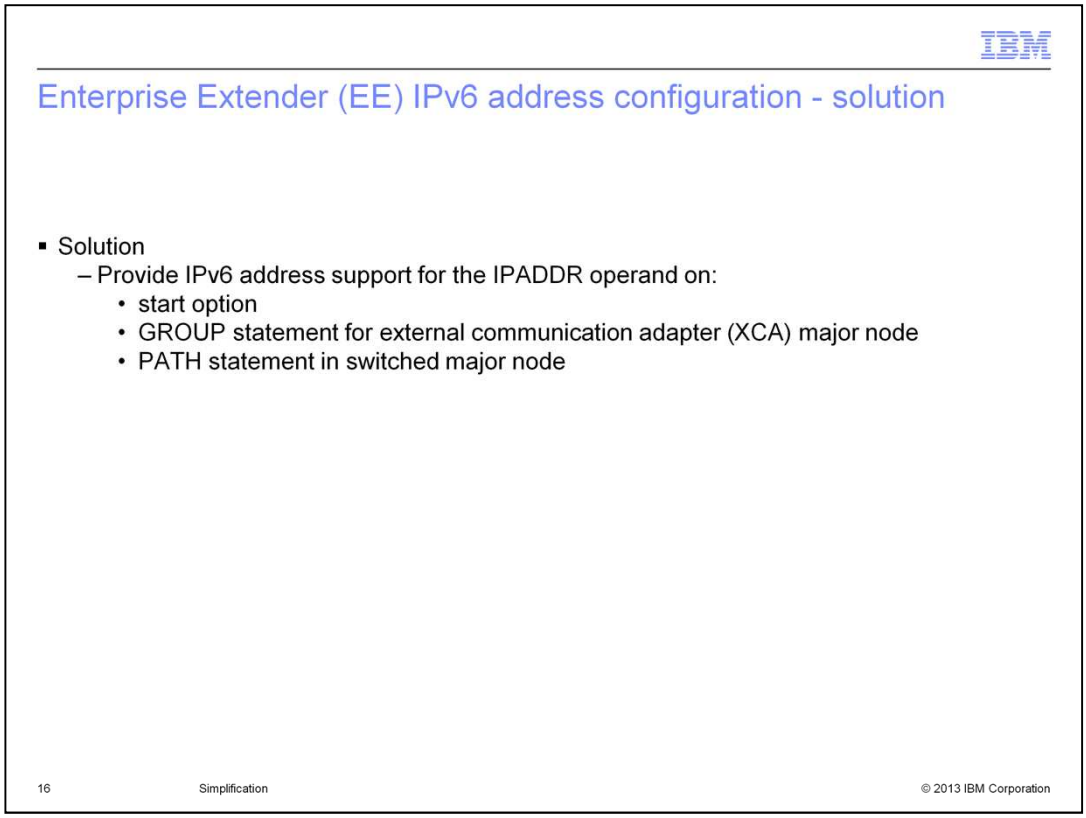
During the many releases since z/OS V1R2, additional resolver setup statements have been introduced and there are multiple statements available to be specified. There is still no need to create a resolver setup file. But, you are more likely to fine-tune the resolver processing, due to functions such as resolver caching and unresponsive name server monitoring. If you don't get the setup statements quite right, the resolver initiation fails. And if the resolver cannot be started, the TCP/IP stacks cannot be started either.



Starting in z/OS V2R1 Communications Server, the resolver now parses the entire setup file during initialization, even if errors are encountered in the file. The resolver still issues messages identifying specific errors in the file. This allows you to use a single resolver setup file for resolver initialization on all systems, regardless of release level. Since the resolver now parses the entire setup file, it is possible that the resolver will encounter correct and incorrect specifications of the same setup statement. If that occurs, the resolver will use the last correct specification, ignoring any previous specifications and subsequent incorrect specifications. If no correct specification is found, the default setting is used.

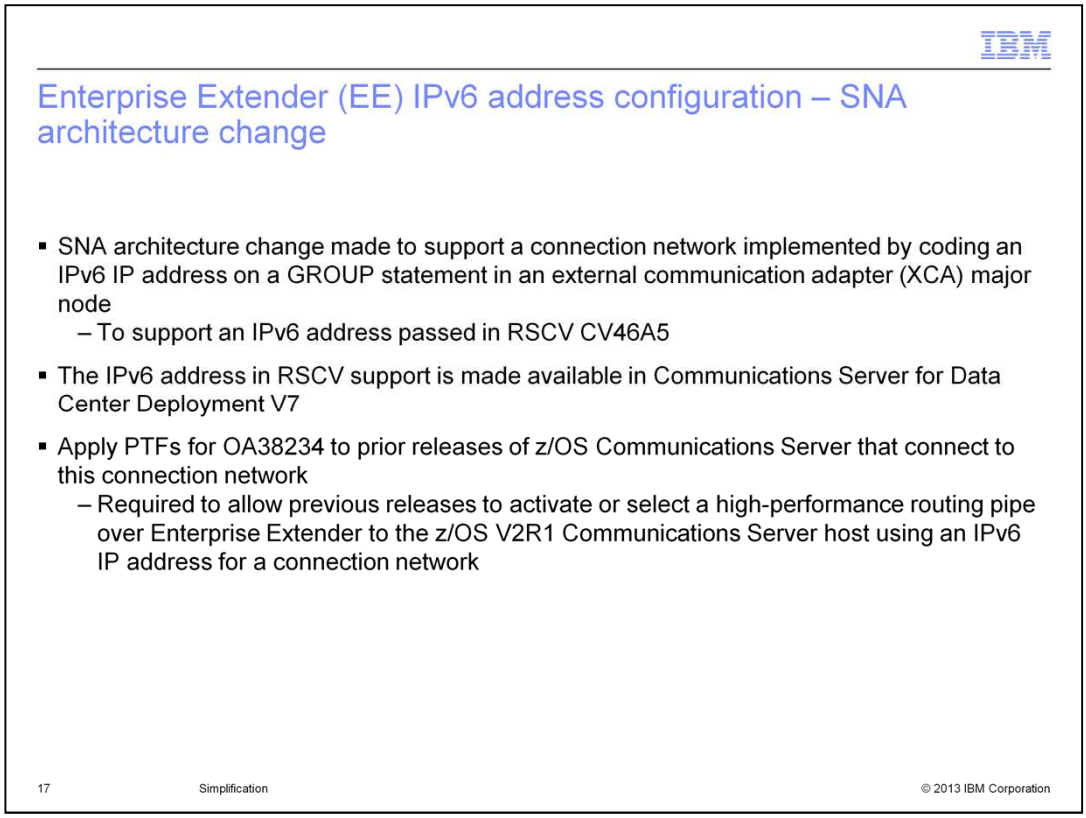


To use Enterprise Extender, VTAM needs the local and remote IP addresses. For IPv4, VTAM can learn this by using IPADDR, HOSTNAME or TCPNAME for the local IP address. For IPv6, in previous releases, IPADDR cannot be used. But HOSTNAME resolution does not always work. And, when TCPNAME is used for the local address, it only returns an IPv4 address from the TCP/IP stack.



z/OS V2R1 Communications Server provides IPv6 address support for the IPADDR operand in three new locations. These are the three remaining places where an IPv4 address is required in the previous release. The first location is the IPADDR modifiable start option, which specifies the local IP address. The second location is the IPADDR on GROUP in the external communication adapter major node, which also specifies the local IP address. The third location is the IPADDR on PATH in the switched major node, which specifies the remote IP address.

There are no changes to the displays. They support an IPv6 address for IPADDR.

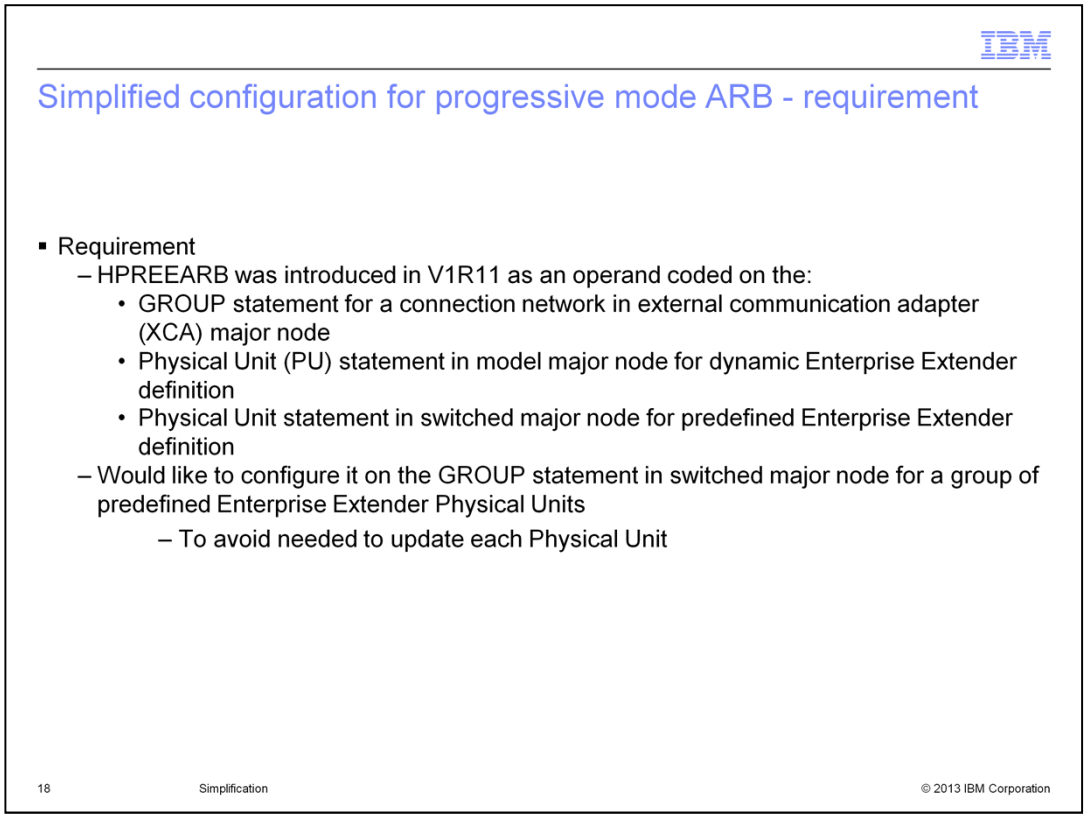


The SNA architecture now passes the IPv6 IP address in RSCV CV46A5. This change was needed in order for a connection network GROUP in the external communication adapter major node to have an IPv6 IP address coded or sifted from the IPADDR start option

Distributed Communications Servers or other high-performance routing products can use a high-performance routing pipe over Enterprise Extender to the z/OS V2R1 Communications Server host using an IPv6 IP address for a connection network. To do this, those products must support receiving an IPv6 address in RSCV for a high-performance routing pipe using connection network. If the destination high-performance routing platform does not support receiving an IPv6 address in RSCV, the high-performance routing pipe activation or selection will fail.

Support for the IPv6 address in RSCV is available in the Communications Server for Data Center Deployment V7 product.

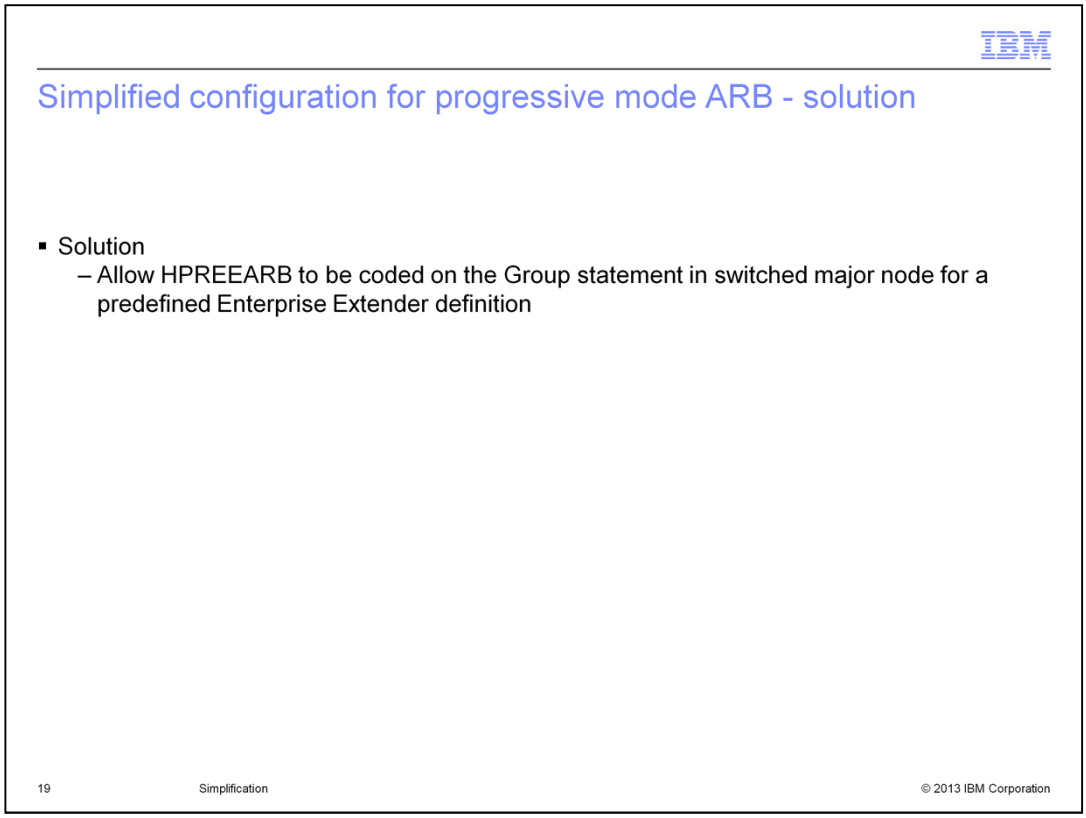
You can implement a connection network by configuring an IPv6 IP address on a GROUP statement in an external communication adapter major node. If you do, you must apply PTFs for OA38234 to prior releases of z/OS Communications Server that connect to this connection network. This is required to allow previous releases to activate or select a high-performance routing pipe over Enterprise Extender to the z/OS V2R1 Communications Server host using an IPv6 IP address for a connection network.



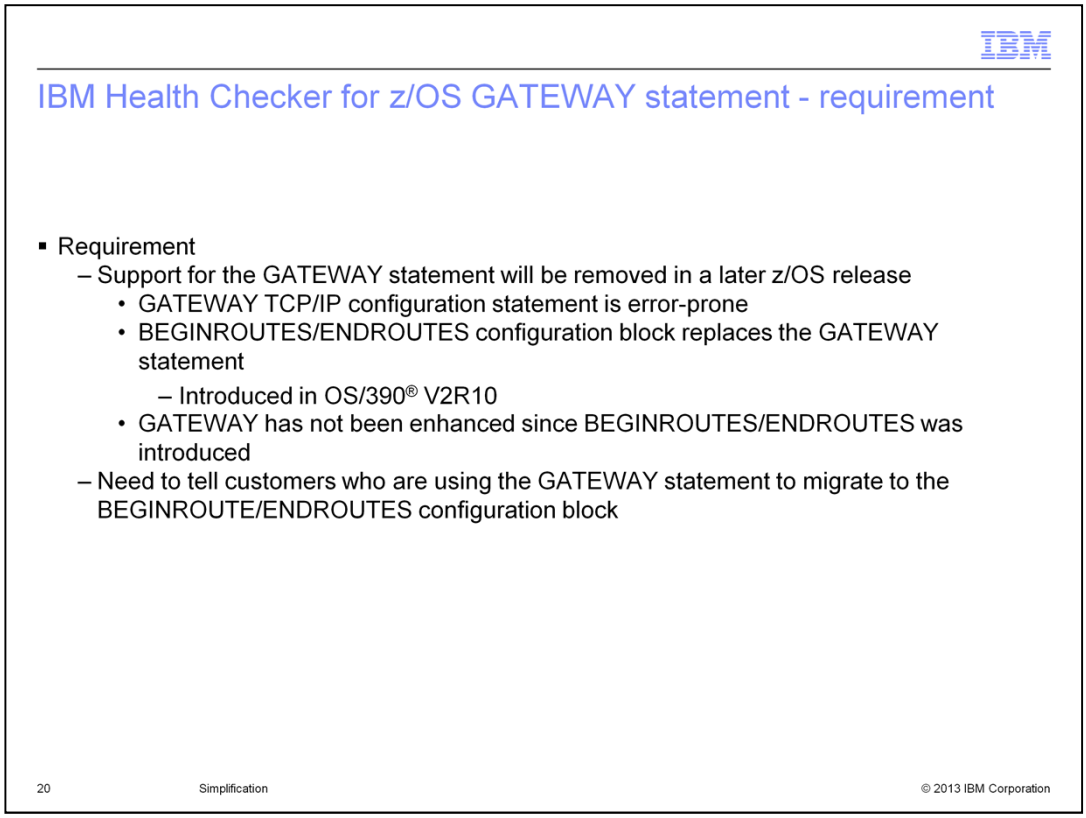
The HPREEARB operand was introduced in V1R11. In releases before V2R1, it can be configured in three places. It can be configured on the Group statement for a connection network definition in an external communication adapter major node. And it can be configured on the Physical Unit statement in either a model major node for a dynamic Enterprise Extender definition or in a switched major node for a predefined Enterprise Extender definition.

HPREEARB allows an Enterprise Extender connection to use a progressive mode adaptive-rate-based flow-control algorithm, if both sides of the Enterprise Extender high-performance routing pipe allow it. Progressive mode adaptive-rate-based is a high-performance routing flow-control algorithm that improves the performance of high-performance routing in virtualized environments. If adaptive-rate-based negotiation finds that progressive mode adaptive-rate-based is not allowed on both sides, the negotiation will typically result in responsive mode adaptive-rate-based.

If you allow dynamic Enterprise Extender Physical Units, then you only need to update a single model Physical Unit to change the adaptive-rate-based value. However, if you predefine your Enterprise Extender Physical Units, then you must update each Physical Unit.



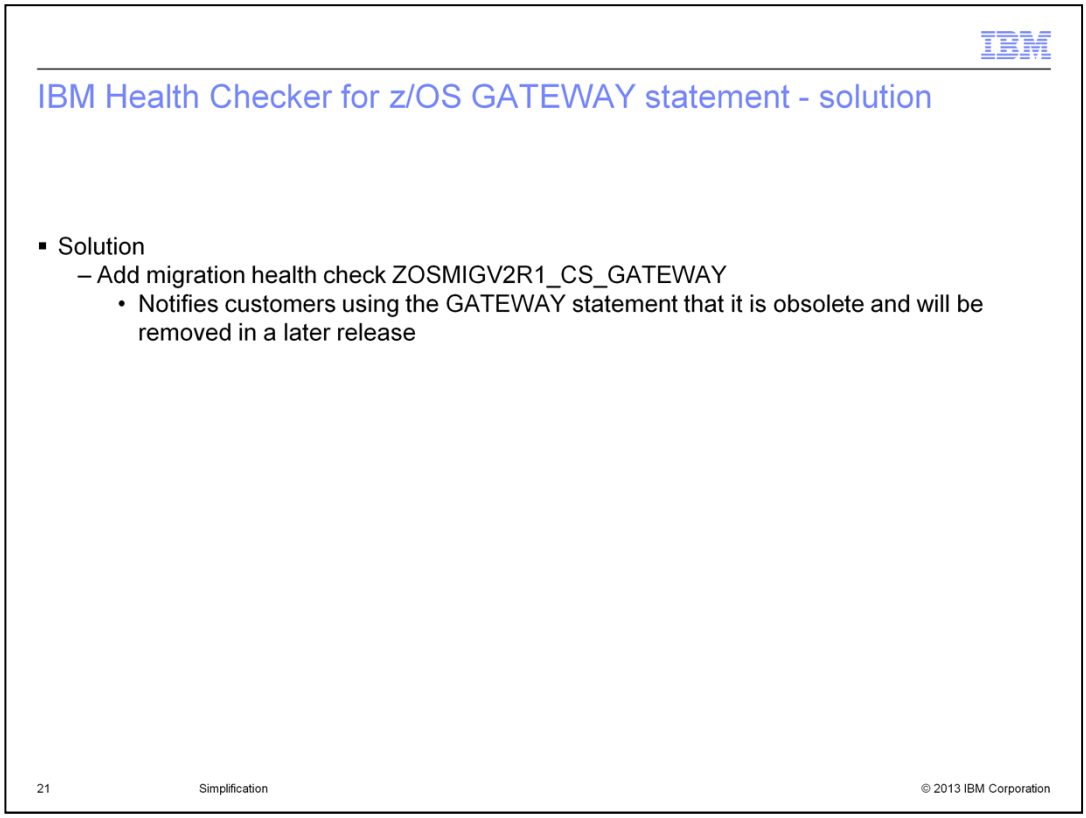
Starting in V2R1, HPREEARB can be configured on the Group statement in a switched major node for predefined Enterprise Extender Physical Units. This allows you to change their predefined Enterprise Extender definitions by making one change, instead of changing each Physical Unit.



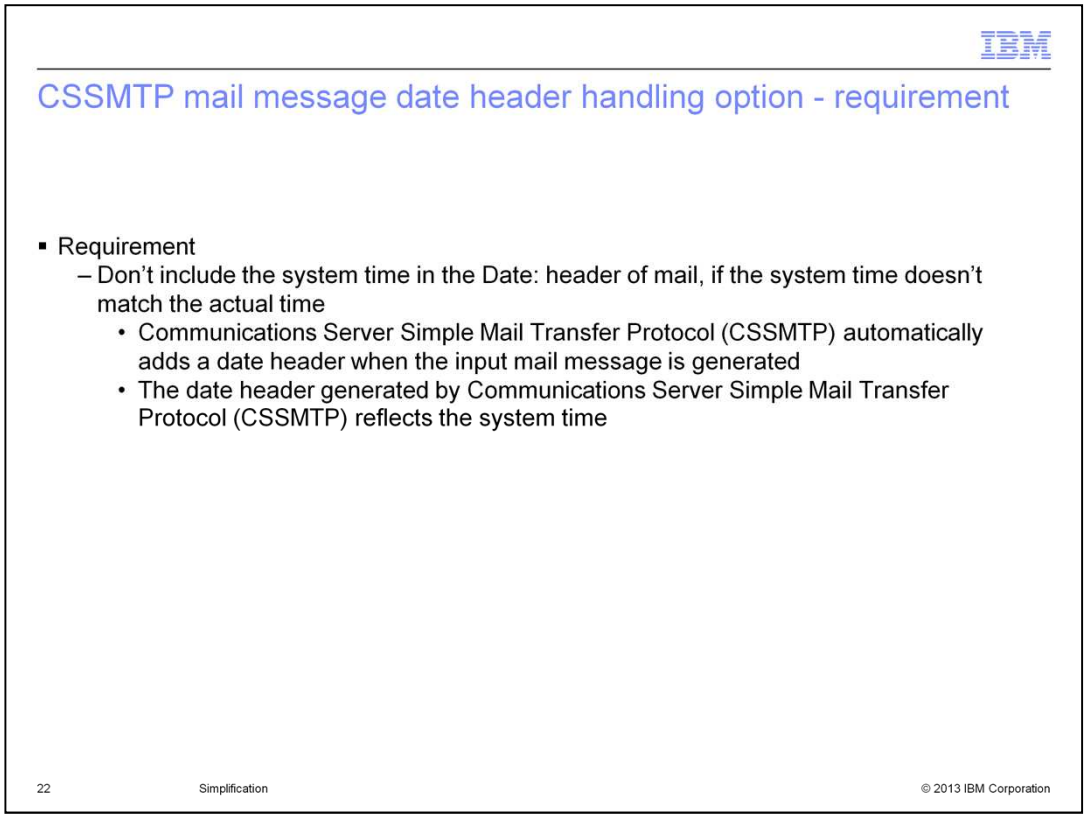
Migration health checks were introduced in z/OS V1R10. They check for potential migration actions when upgrading to a new z/OS release. Migration checks have a naming convention that allows them to be activated or deactivated when you are preparing and planning for migration. They are not enabled by default. You can activate a specific check or use a wildcard check name to activate more than one check. Checks are built into the component code using the z/OS Health Checker infrastructure and checks will generate messages that indicate which migration actions are required.

The GATEWAY TCP/IP configuration statement is error-prone. The more straightforward BEGINROUTES/ENDROUTES configuration block that was introduced in OS/390 V2R10 was intended to replace the GATEWAY statement. The GATEWAY statement has not been enhanced since BEGINROUTES/ENDROUTES was introduced and therefore does not support features like IPv6, replaceable static routes, and so on.

Support for the GATEWAY statement will be removed in a later z/OS release. Therefore customers who are currently using GATEWAY, need to be told to migrate to BEGINROUTES/ENDROUTES.

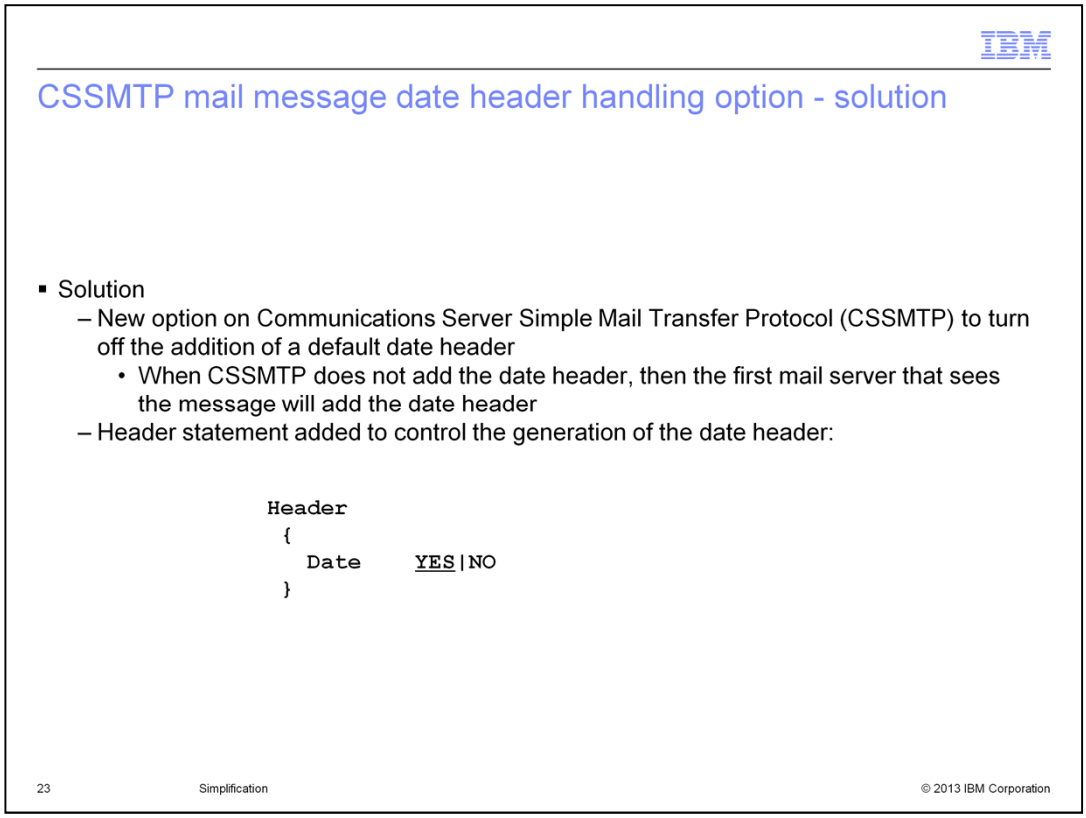


In z/OS V2R1, a new migration health check notifies customers who are still using the GATEWAY statement that it is obsolete and will be removed in a later release. The new migration health check is named ZOSMIGV2R1_CS_GATEWAY.

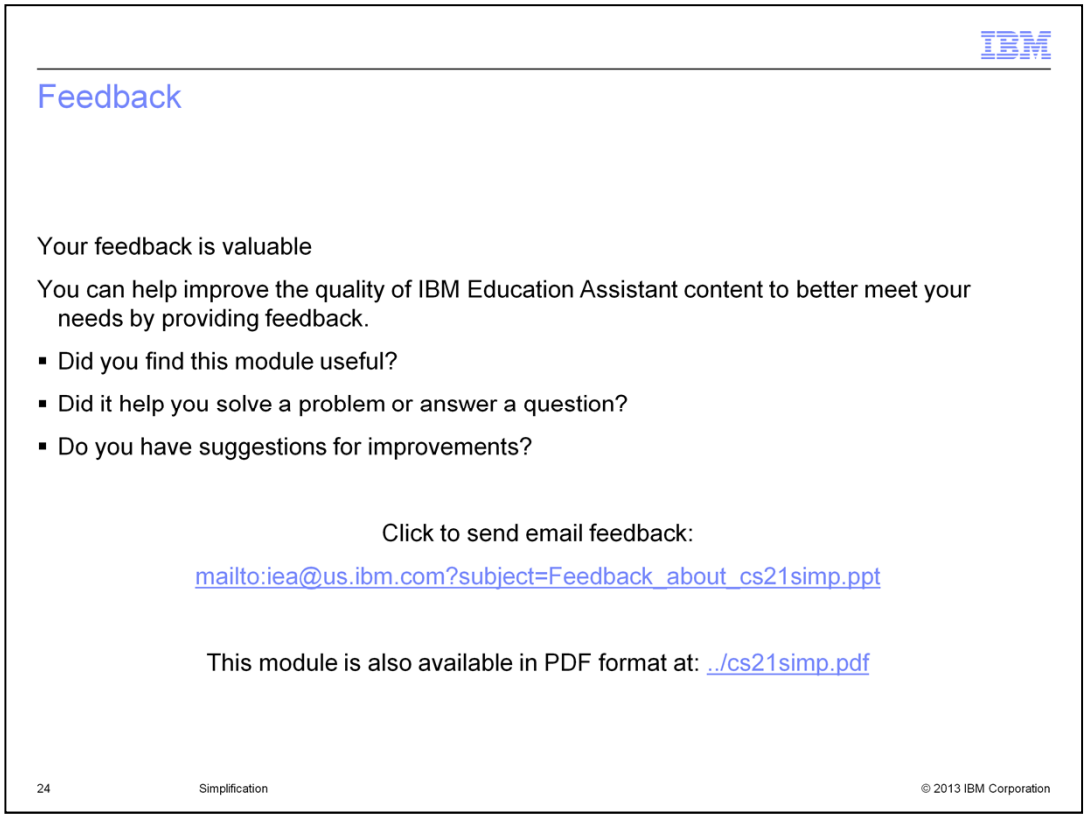


Communications Server Simple Mail Transfer Protocol inserts a Date: header into each mail that does not have one. It is the system time that is inserted. This is undesirable if the system time does not reflect the actual time, for example, if the system time is set back by twelve hours in order to run production work for the previous day.

Communications Server Simple Mail Transfer Protocol implements RFC 2822 standards.



In z/OS V2R1, Communications Server Simple Mail Transfer Protocol now accepts a new statement, called Header, and a new parameter, called Date, to control the creation of the Date: header. If Header Date NO is configured, then Communications Server Simple Mail Transfer Protocol will not create a date header if one is missing.



You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, disclaimer, and copyright information