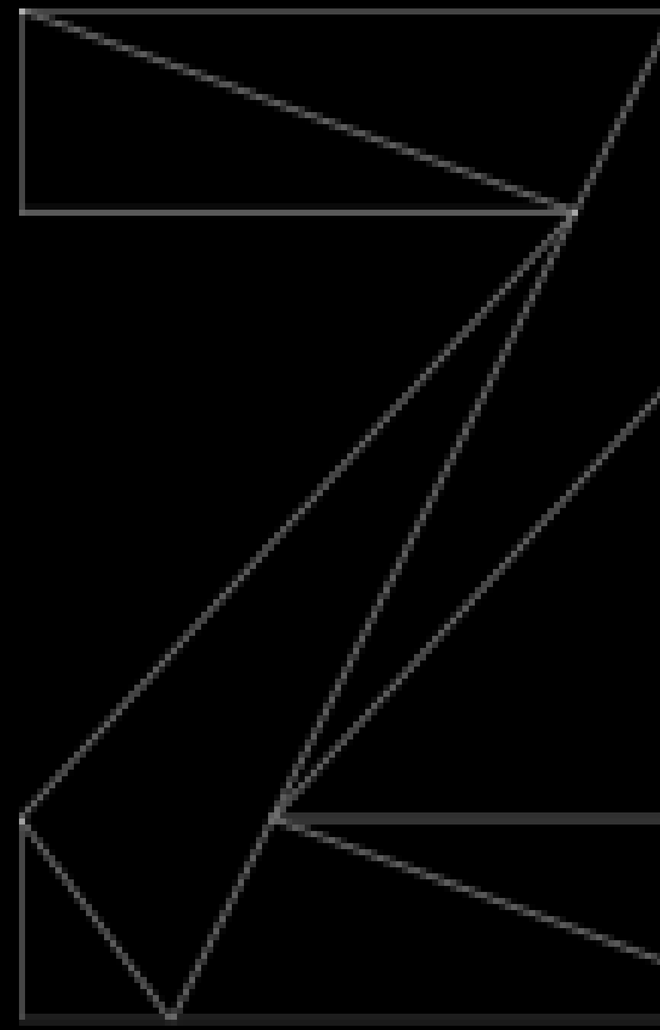
Reusable Configuration and Symbols



IBM Z

Introduction to Reusable Configuration

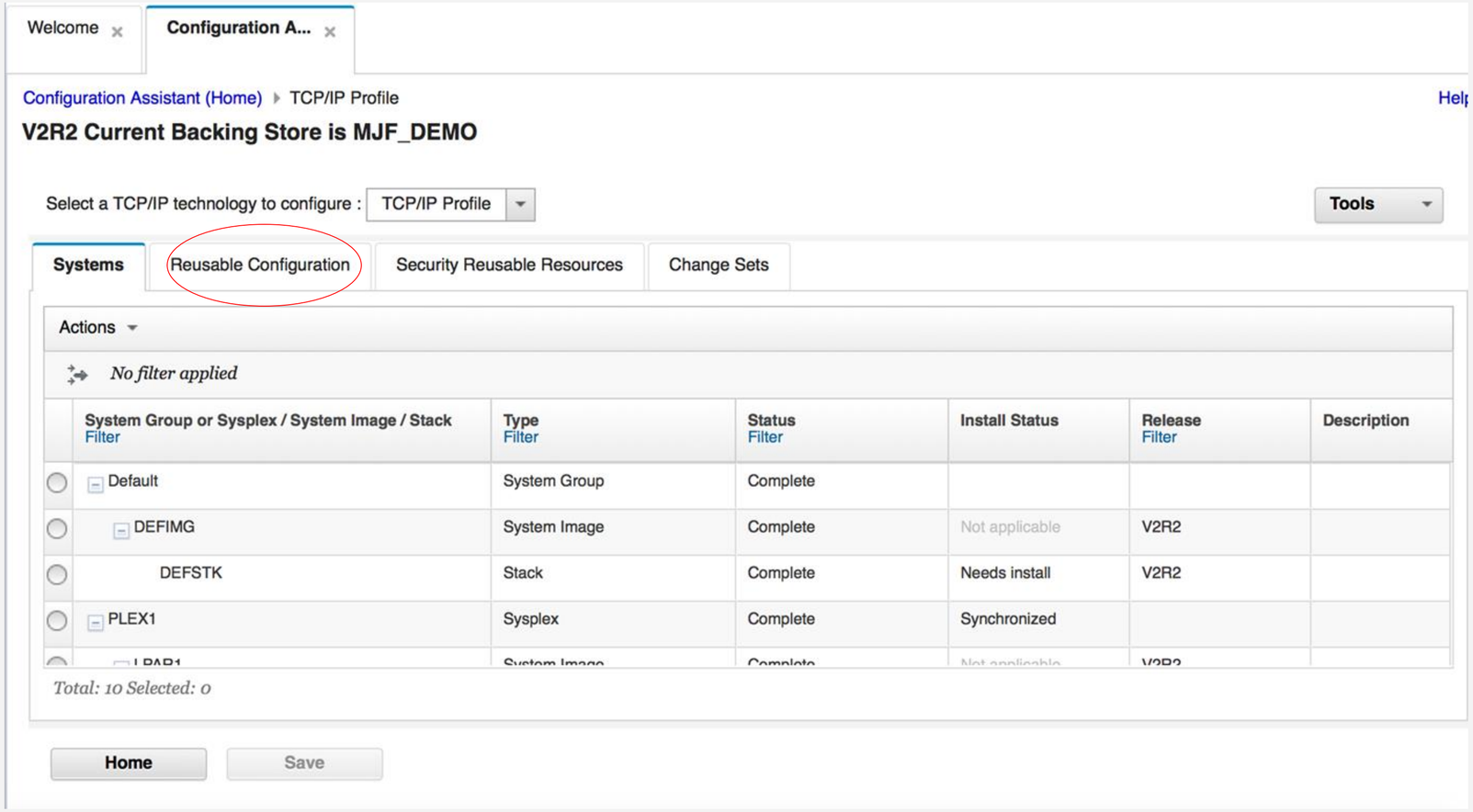
Many customers use INCLUDE files in their TCP/IP profiles to help manage common configuration across multiple stacks.

Examples of INCLUDE file use cases:

- Common PORT and AUTOLOG definitions
- Common INTERFACE definitions
- Common configuration options
  - For example: GLOBALCONFIG, IPCONFIG, etc.

In Network Configuration Assistant, this is handled using objects known as **Reusable Configuration.**

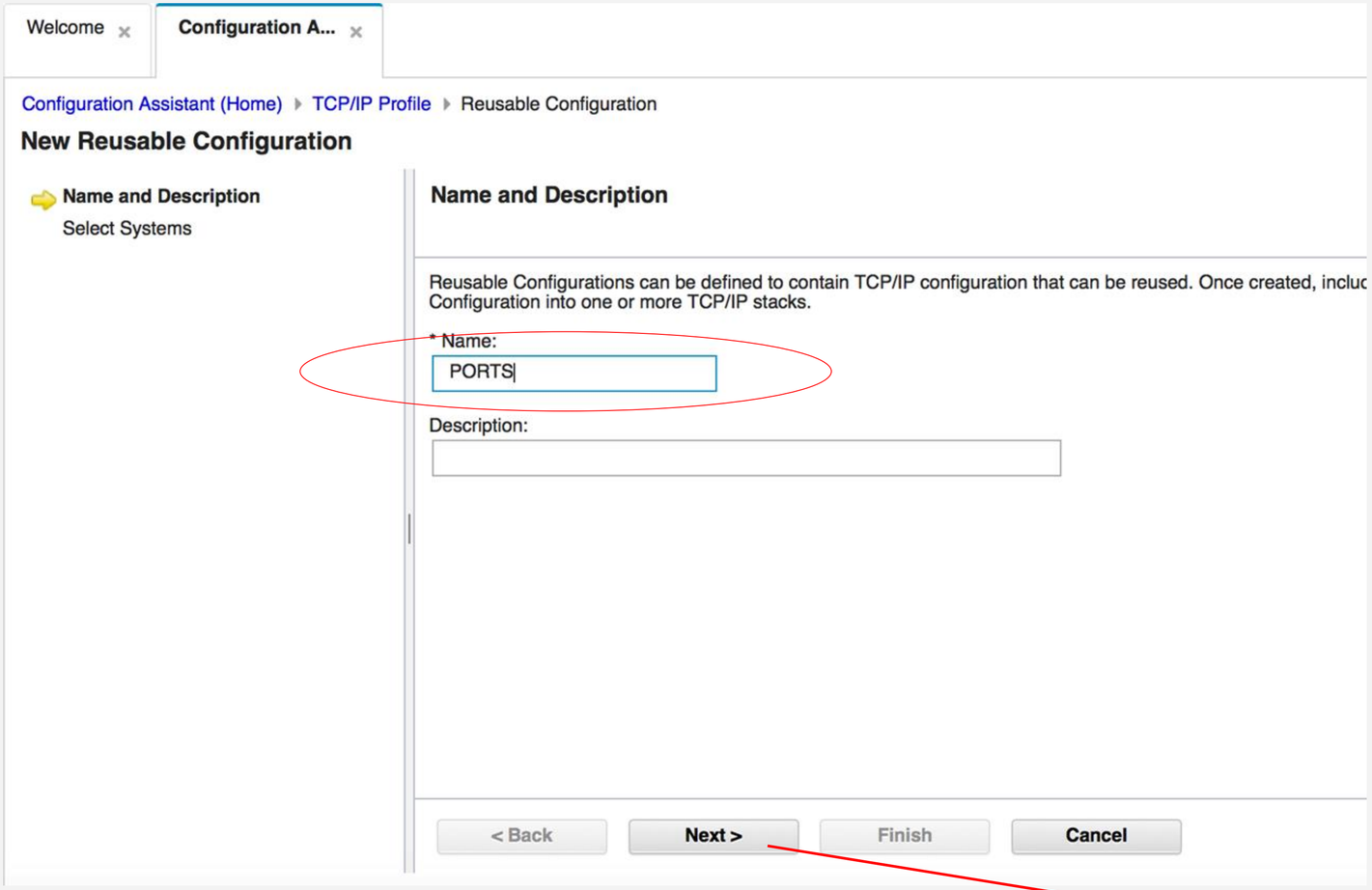
Reusable configuration, continued



Reusable configuration objects are accessed from their own tab on the main TCP/IP systems tree.

A reusable configuration object is a set of configuration that will be used by multiple TCP/IP stacks in the Network Configuration Assistant.

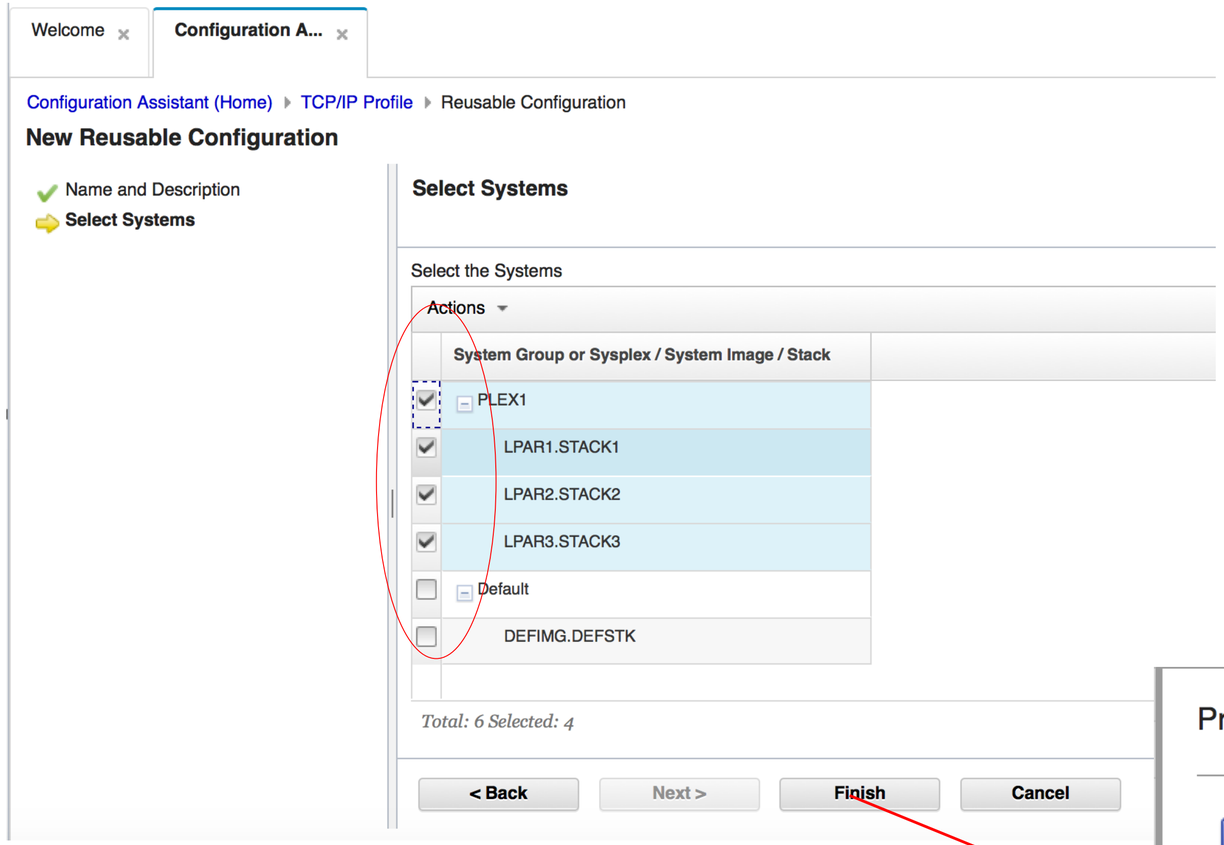
Creating reusable configuration



In this example we will create a reusable configuration for common port definitions. First you give the RC (Reusable Configuration) a name...

Next page

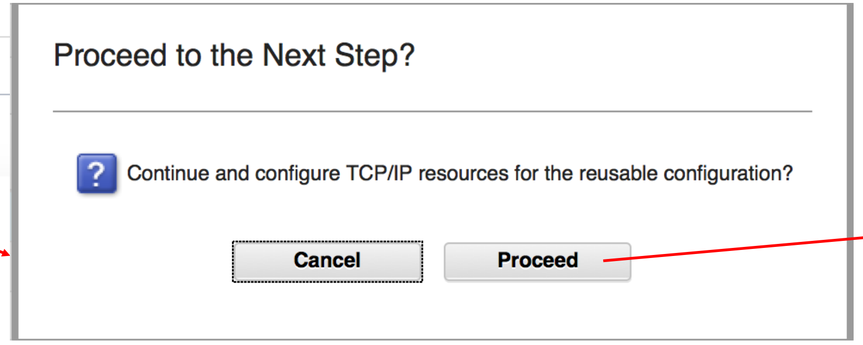
Creating reusable configuration, cont.



Then you select which stacks will use this RC. These are the stacks that will make use of the common configuration that you create in the RC.

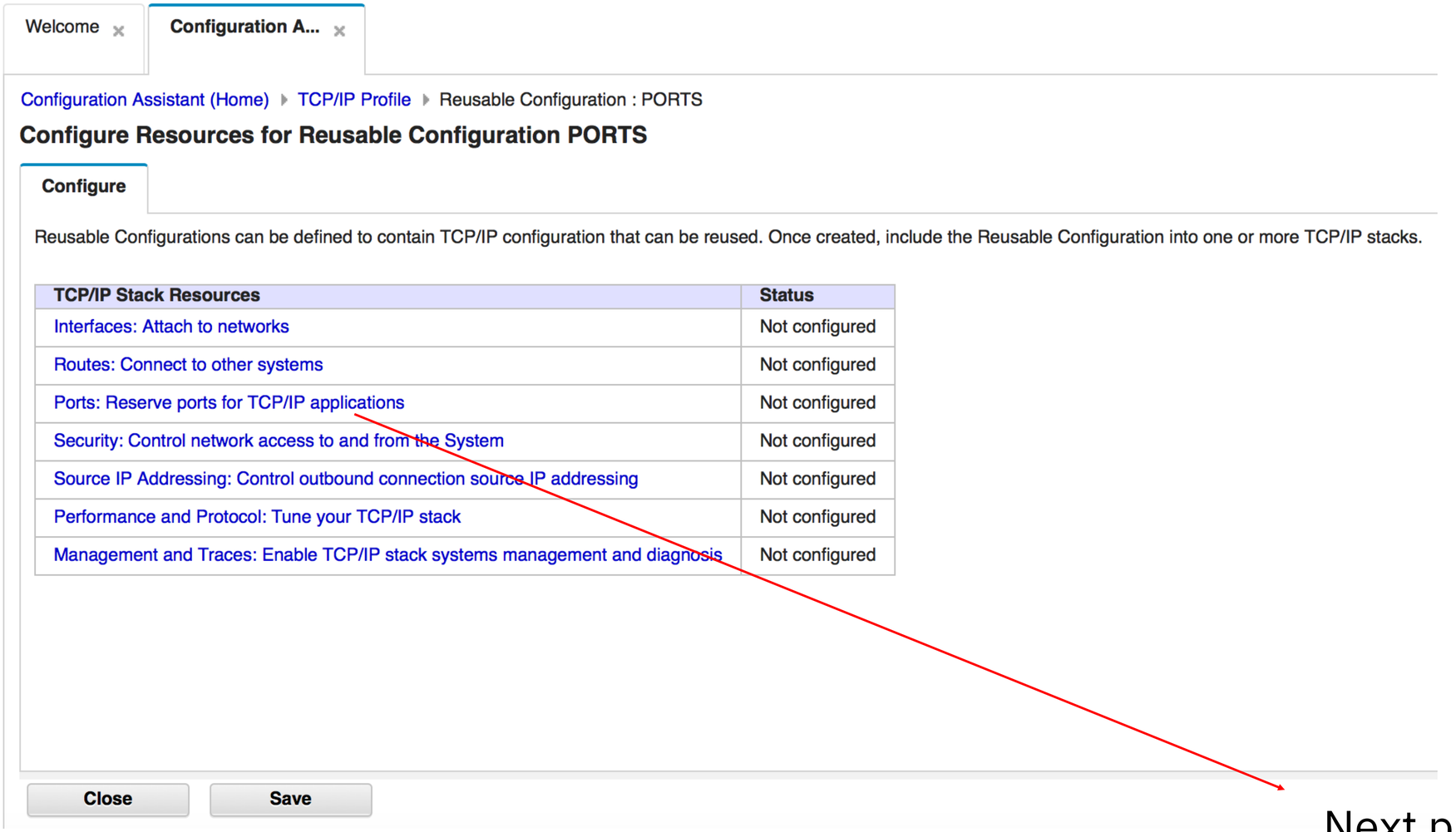
In this example, we are creating common PORT definitions that will be used by all the stacks in PLEX1.

Once the RC is created we are prompted to configure it.



Next page

Configuring reusable configuration



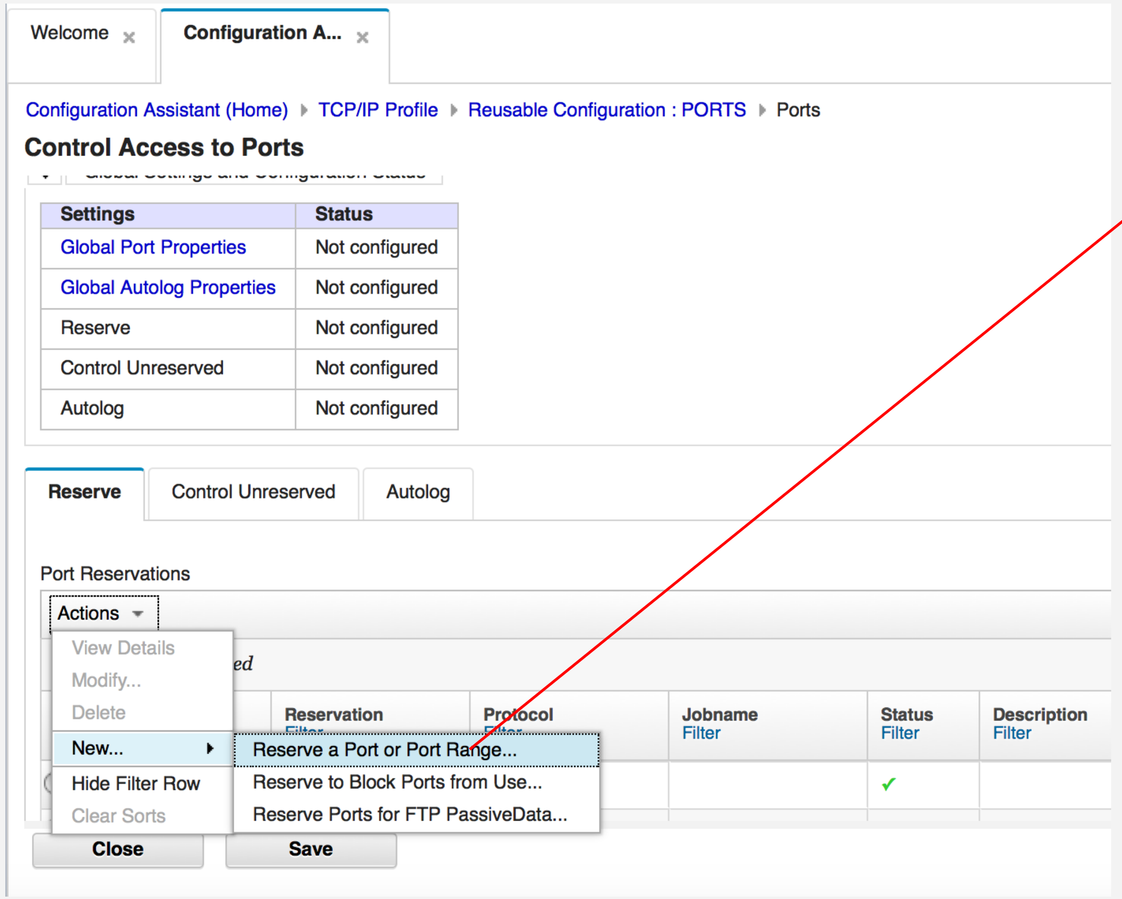
You configure reusable configuration just like any other TCP/IP configuration, using the same panels.

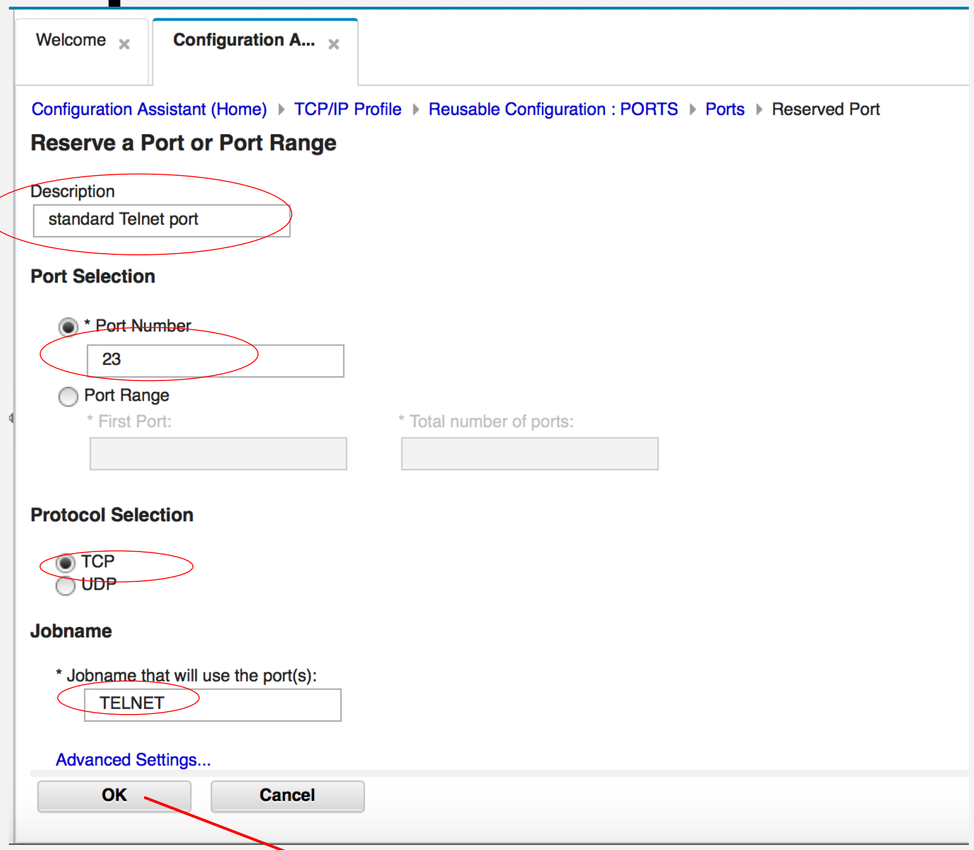
- There are some exceptions that will be touched on later

In this case we are configuring common ports, so we select port definitions from the main resource screen.

Next page

RC Port configuration example

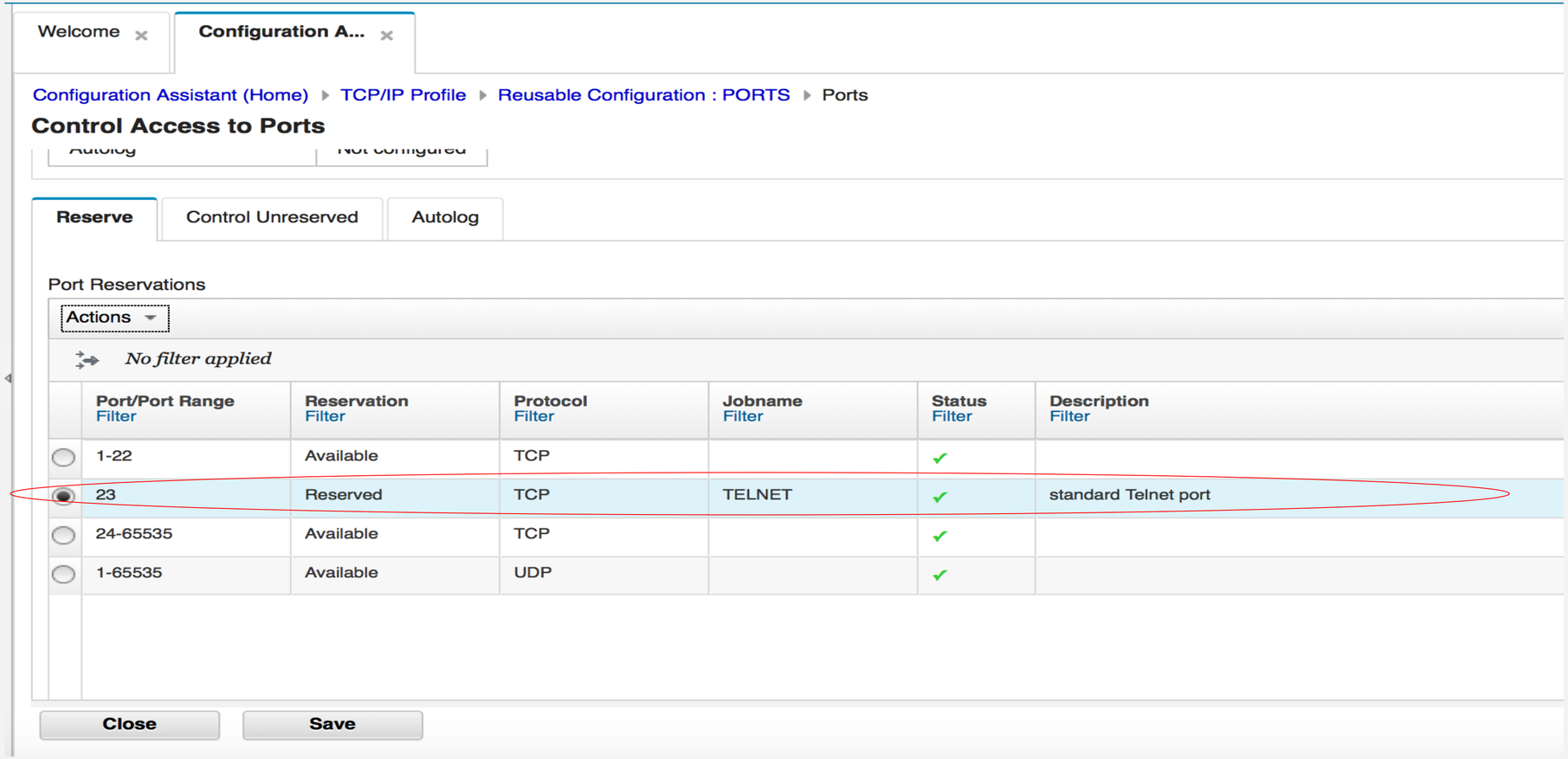




In this example, we have created a reservation for the standard telnet port.

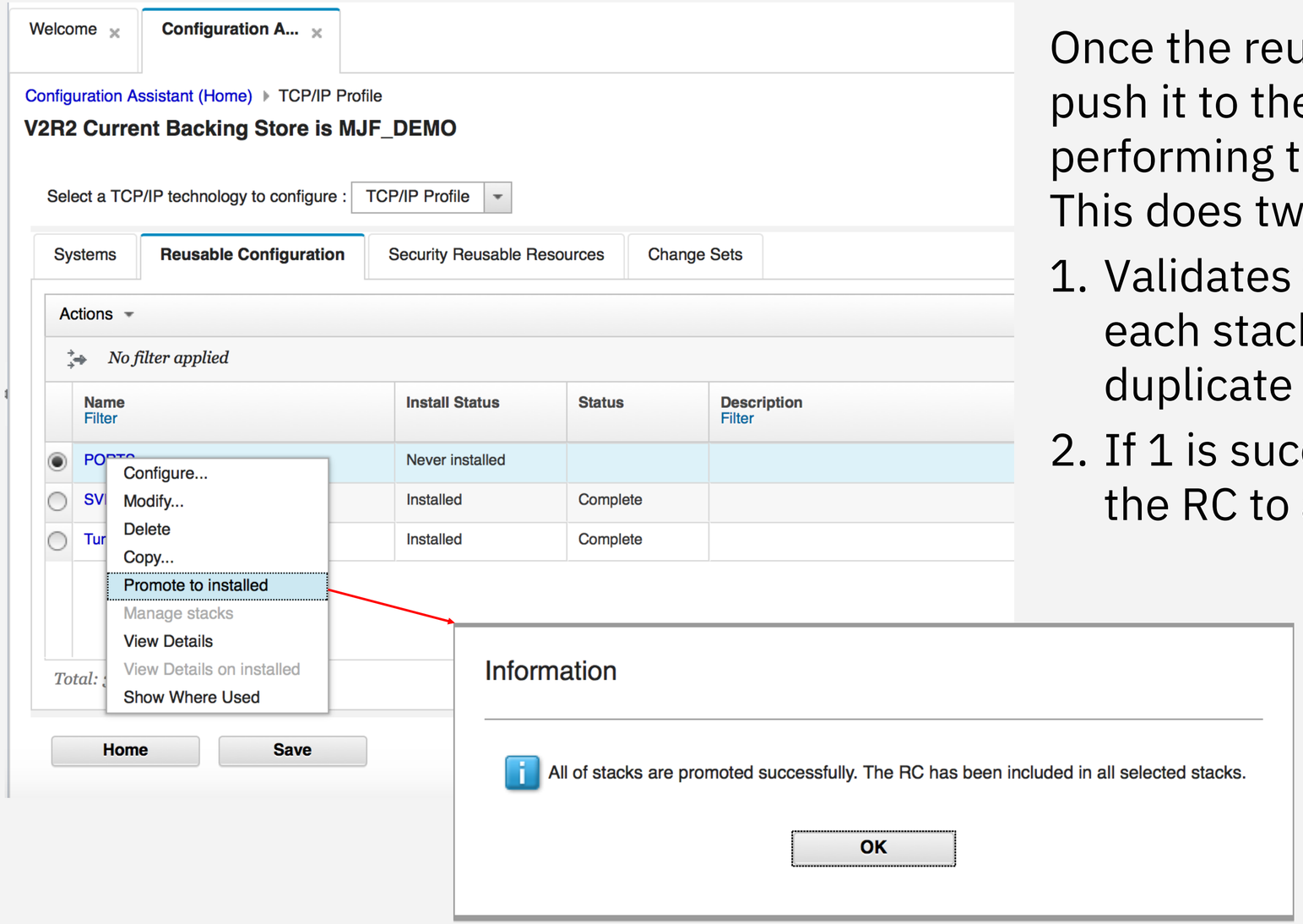
Next page

RC Port configuration example



In this example, we have created a reservation for the standard telnet port in this reusable configuration.

Push the reusable configuration to stacks

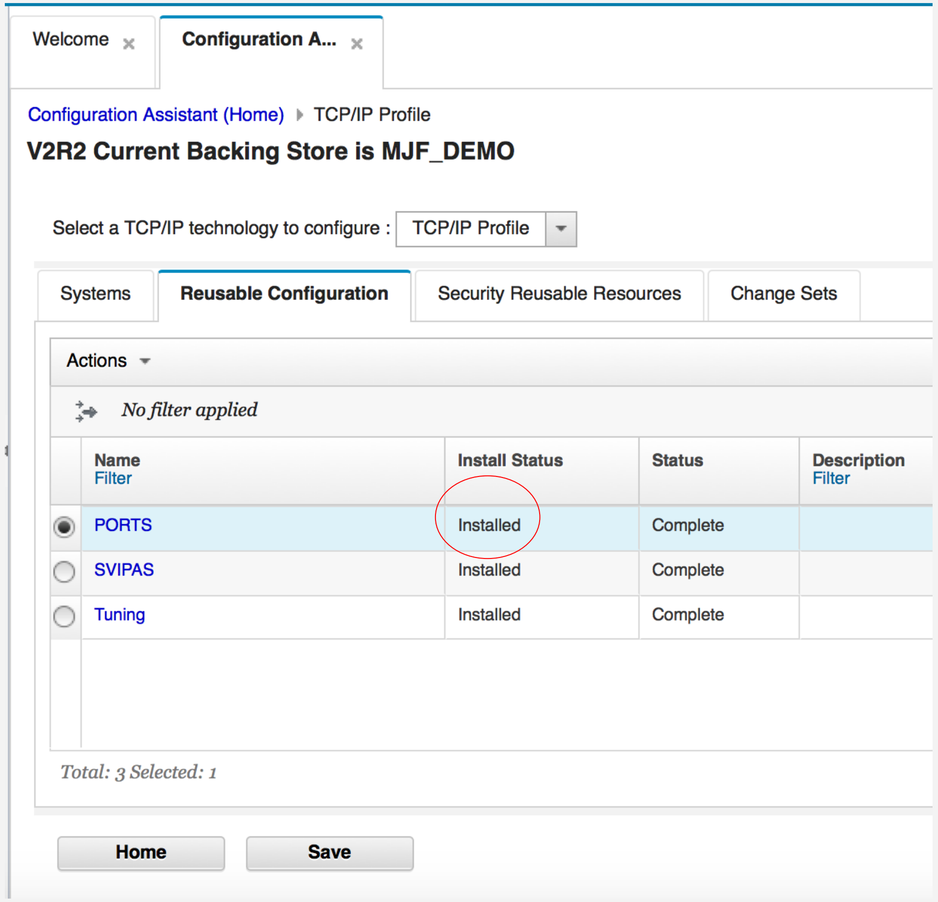


Once the reusable configuration is created, you push it to the stacks that use this RC by performing the “Promote to installed” action. This does two things:

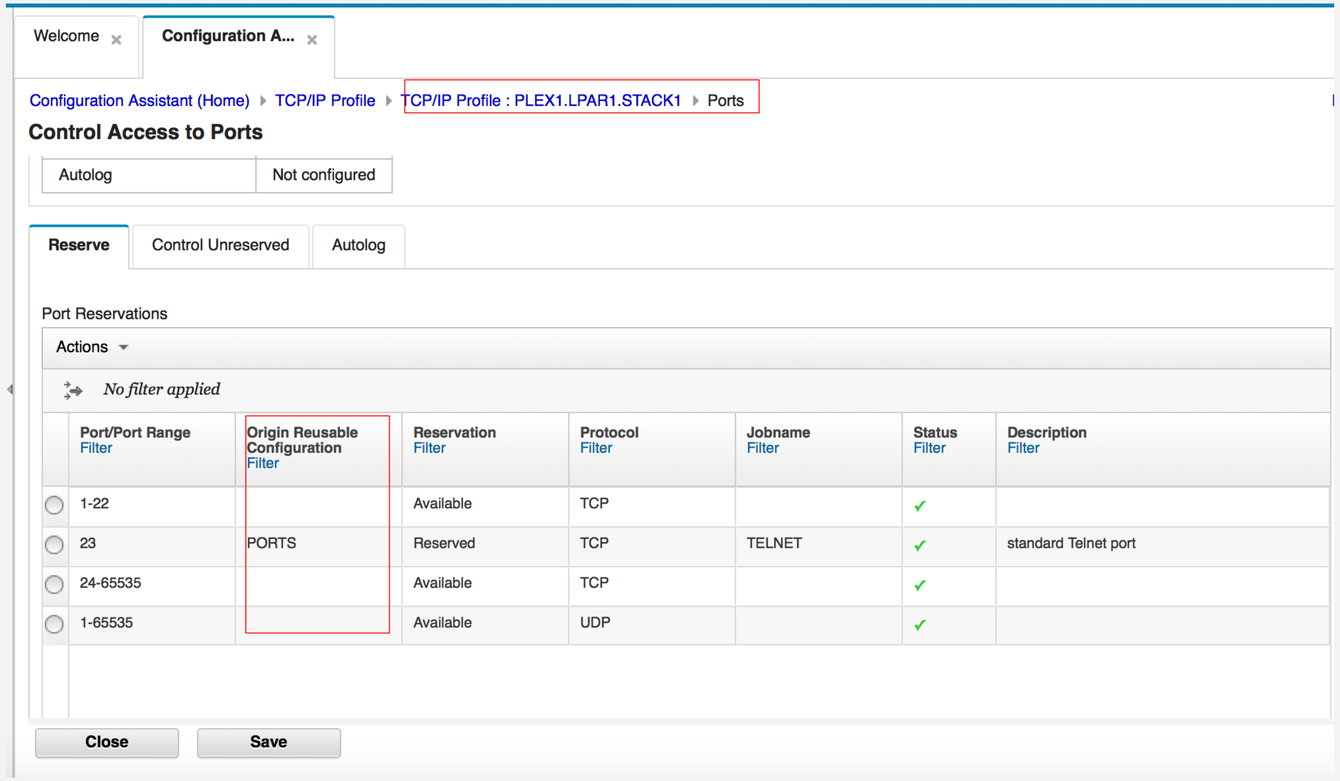
1. Validates the reusable configuration against each stack (e.g., ensures no conflicts like duplicate names)
2. If 1 is successful, adds the configuration from the RC to all the using stacks.

Note: reusable configuration cannot be modified from within the using stacks. It must be modified in the RC interface, and then reinstalled.

Installed reusable configuration

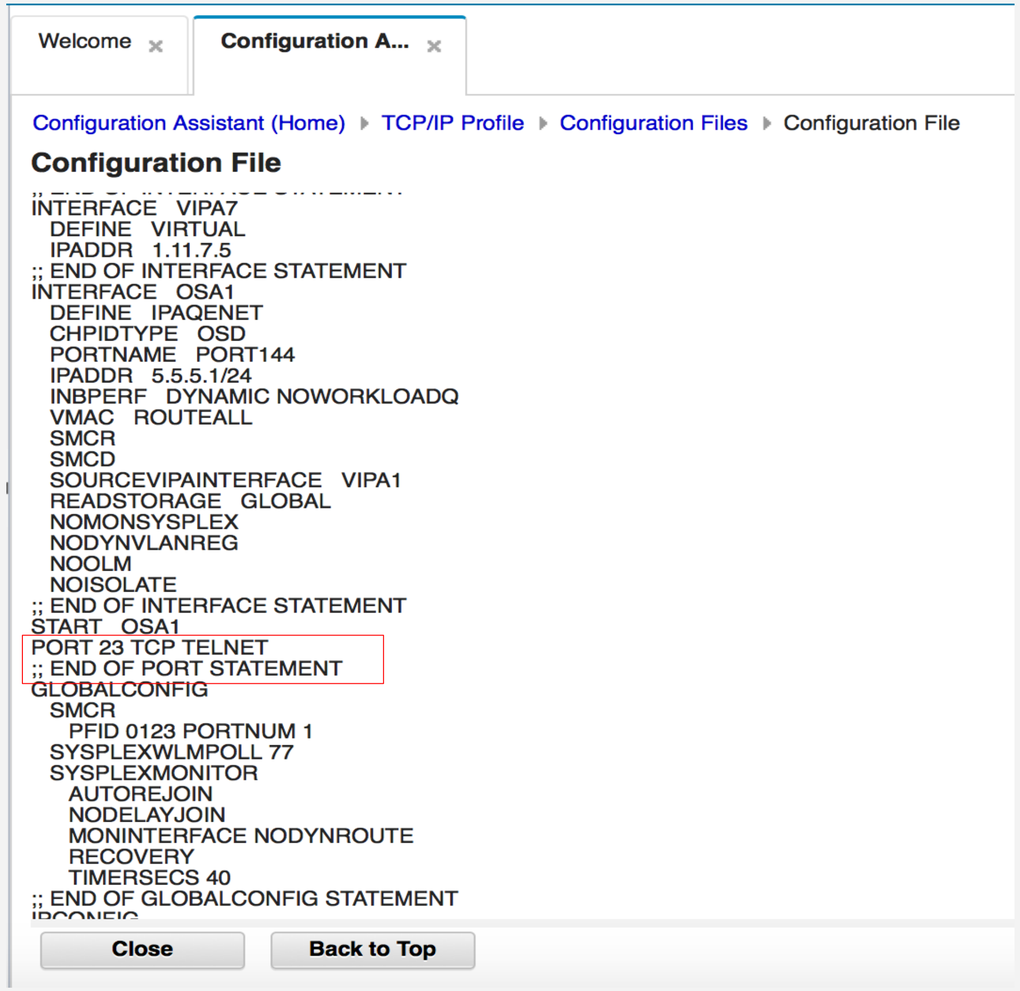


In the **Reusable Configuration** panel, a status of Installed indicates that the contents of the RC have been pushed to all using stacks.



This is the port configuration panel for STACK1. You can see that the port 23 reservation came from reusable configuration PORTS. If you want to modify this definition, you must do it in the RC, not here.

Reusable configuration generated files



Network Configuration Assistant combines all configuration into a single TCP/IP profile when installing files, so the reusable resources are simply included in the profile, not maintained by NCA as separate include files.

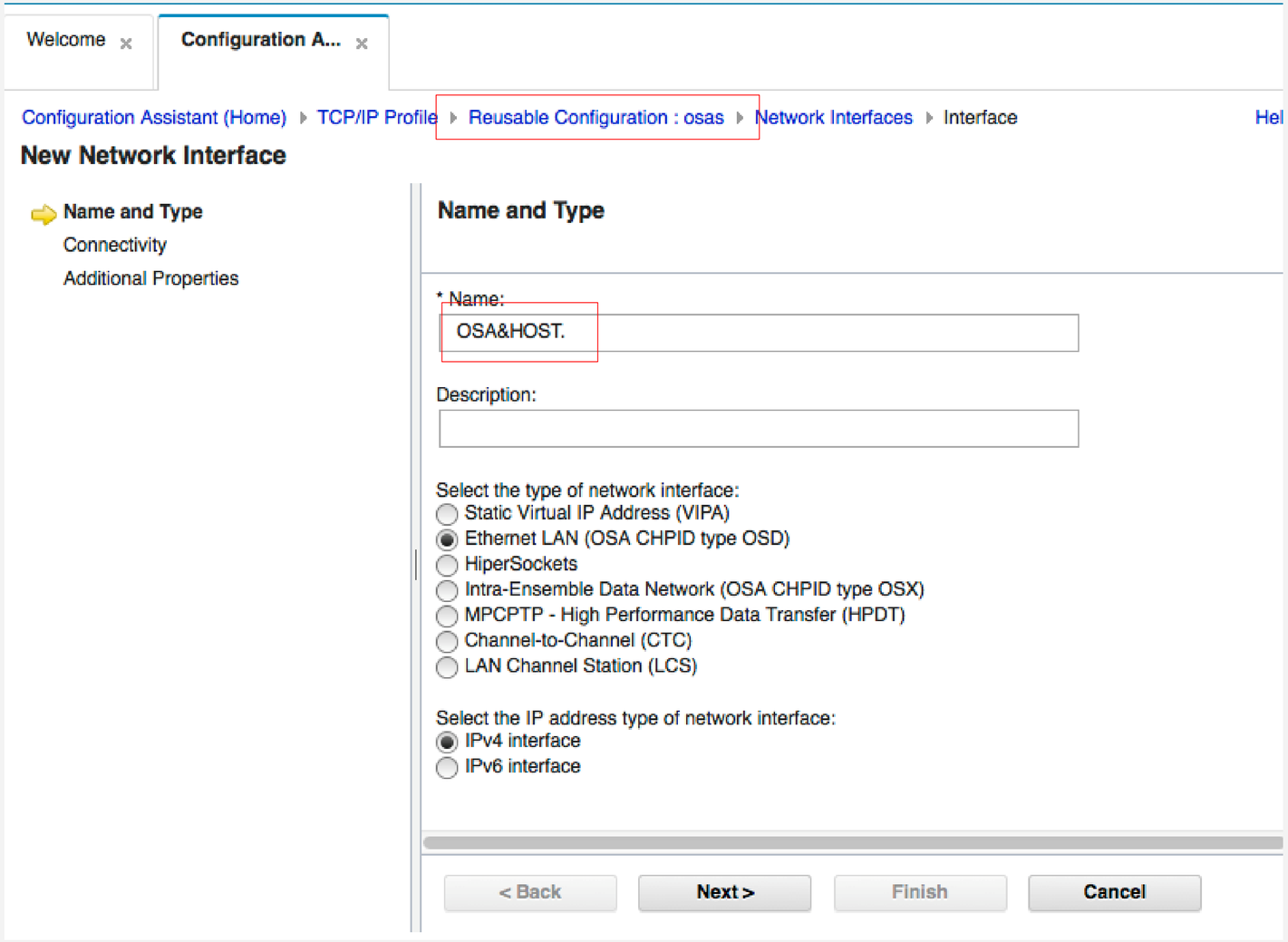
Network Configuration Assistant system symbols

- Customers often use MVS system symbols along with include files to manage common configuration. Common use cases for MVS system symbols include:
  - Interface name and/or IP address for an interface that is in a common include file, for example:
    - 1.1.1.&HOST.
    - OSA&CHPID.
- Network Configuration Assistant implements its own system symbols.
  - NCA system symbols have the same syntax as MVS system symbols (&SYMBOL_NAME.)
  - NCA does not access MVS system symbol values
    - When importing TCP/IP configuration (covered in a later lesson) MVS system symbols are converted to NCA system symbols
  - NCA system symbols are created and resolved within the Network Configuration Assistant and are used when configuring stacks
    - NCA system symbols are **defined in reusable configuration** and then given values for each stack that uses that reusable configuration

Network Configuration Assistant system symbols, continued

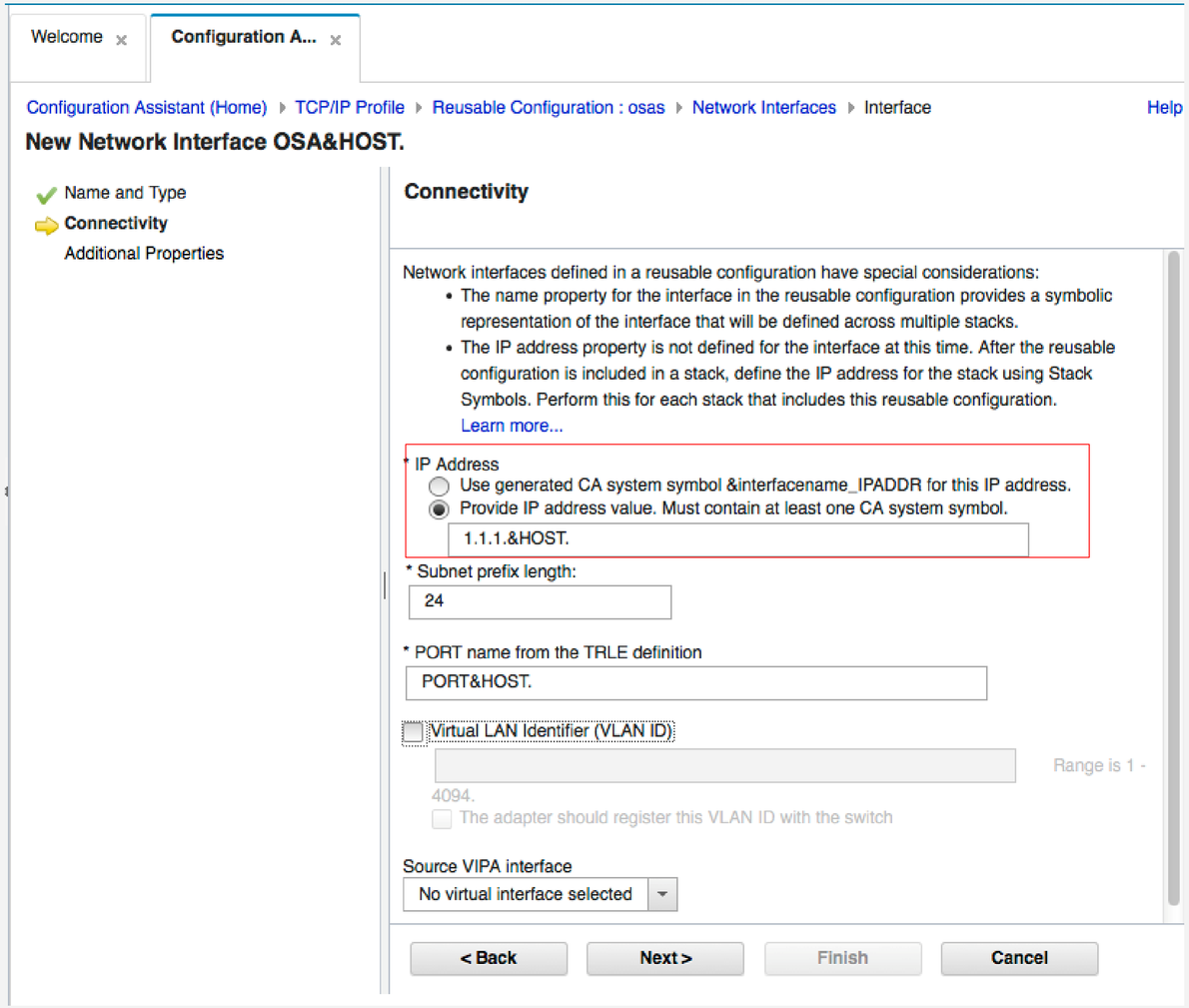
- Network Configuration Assistant then generates TCP/IP profiles with all symbols resolved
  - A stack with unresolved NCA system symbols is incomplete and cannot be installed until all referenced symbols are resolved.
- NCA system symbols are defined in Reusable Configuration
  - Value can vary across stacks that use that Reusable Configuration
- NCA system symbols are supported in the following fields:
  - Interface name
  - Interface IP address
    - Rule: because interfaces can’t have the same IP address across stacks, it is mandatory to use an NCA system symbol in the IP address of any interface that is defined in Reusable Configuration.
  - Interface TRLE port name
  - Interface VLAN identifier
  - SRCIP IP address
  - Static route next hop address (starting with V2R3 APAR PI97737)
- These fields were chosen based on customer use case feedback.
  - If NCA system symbol support is needed in additional fields, please open an RFE against the z/OS Communications Server team

Defining an NCA system symbol



- To define an NCA system symbol, simply enter it into a supported field **in a reusable configuration**.
- In this example, NCA system symbol &HOST. is being created.

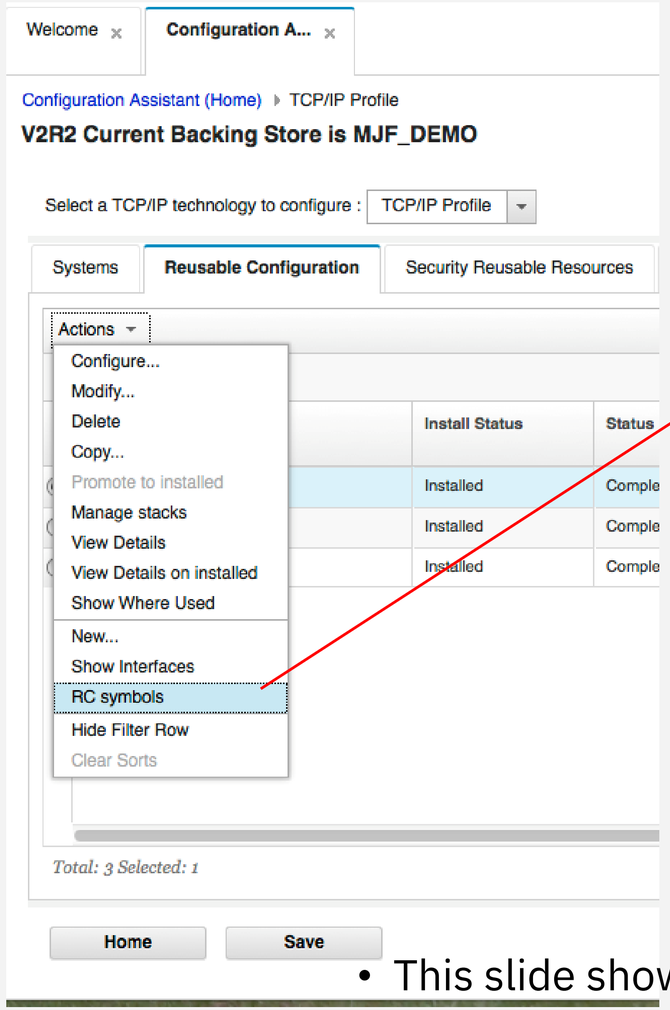
Special symbol rule for reusable interfaces

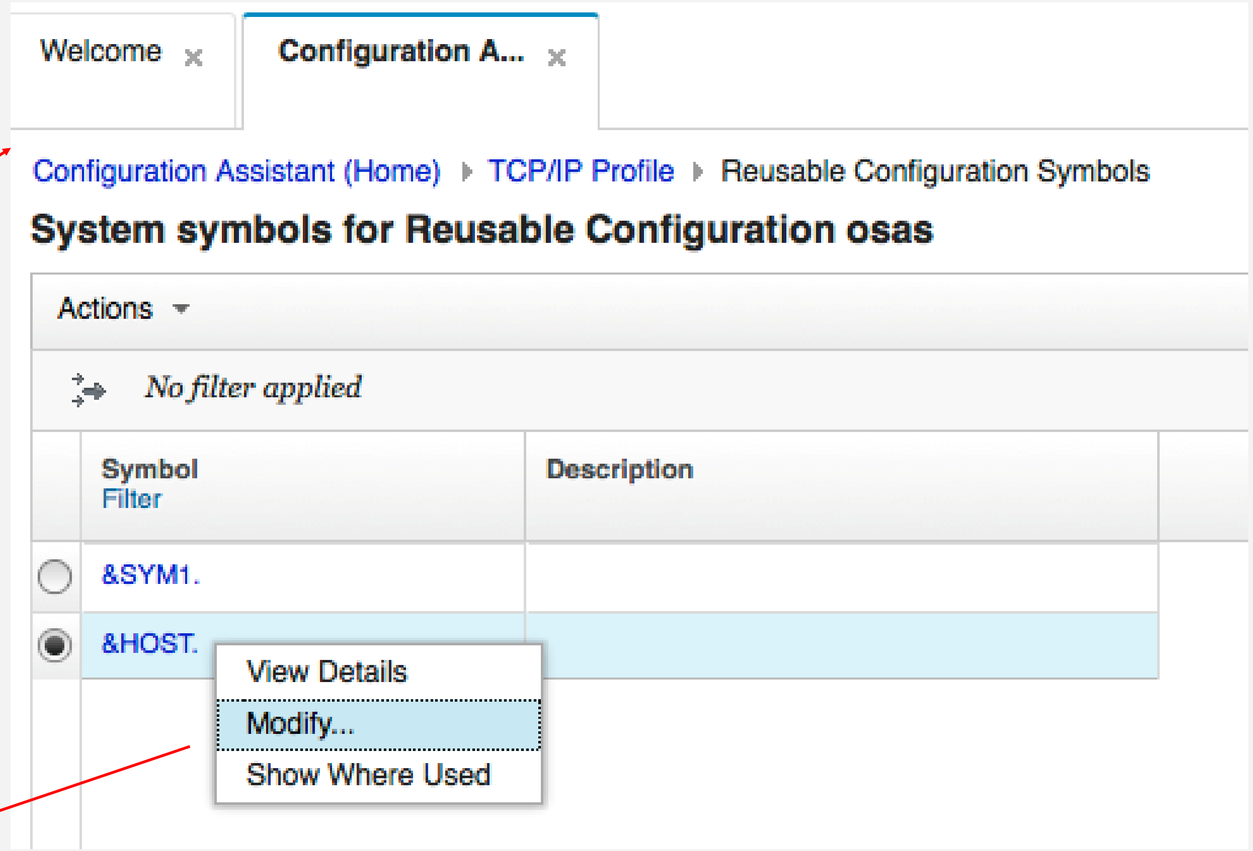


The IP address of an interface defined in reusable configuration must be or contain an NCA system symbol.

- This is because reusable interfaces are meant to be included into multiple stacks, and you can't use the same IP address in multiple stacks! Have to be able to set the IP address per stack.
- You can instruct NCA to create a default IP address symbol, or you can specify the IP address with a symbol.
  - The default IP address symbol for this interface would be &OSAHOST_IPADDR.
- You can specify a new symbol here, or as in this example, reuse a symbol that's already been created.
  - Note that the symbol &HOST. is also used in the TRLE port name

Resolving symbol values: via the RC

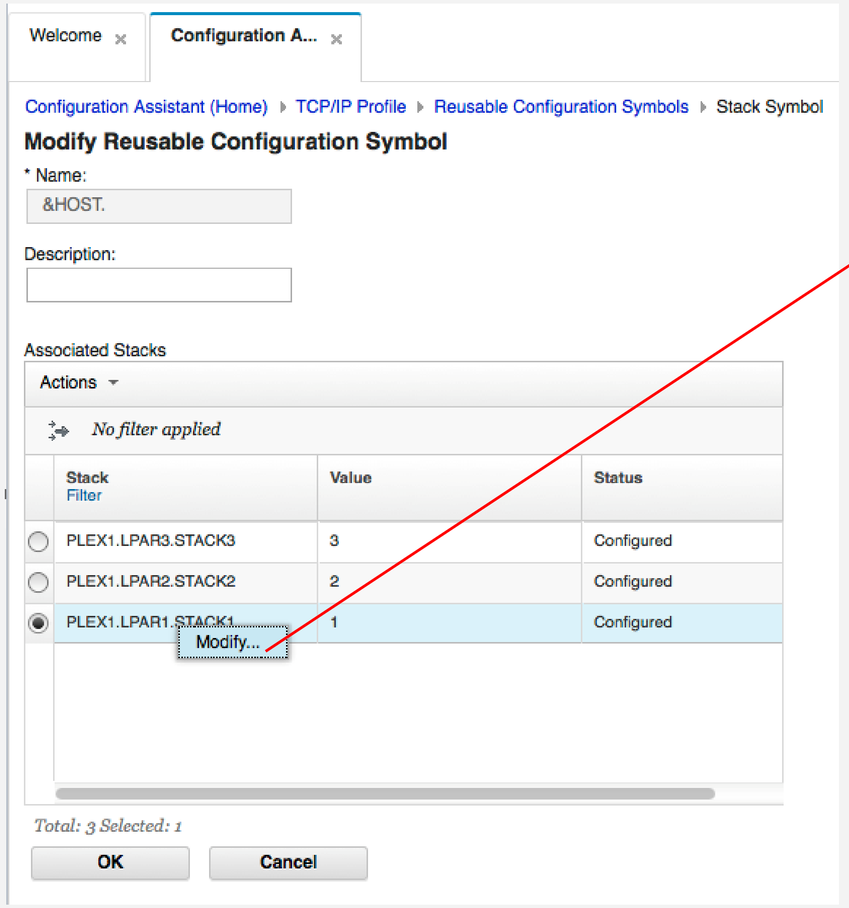


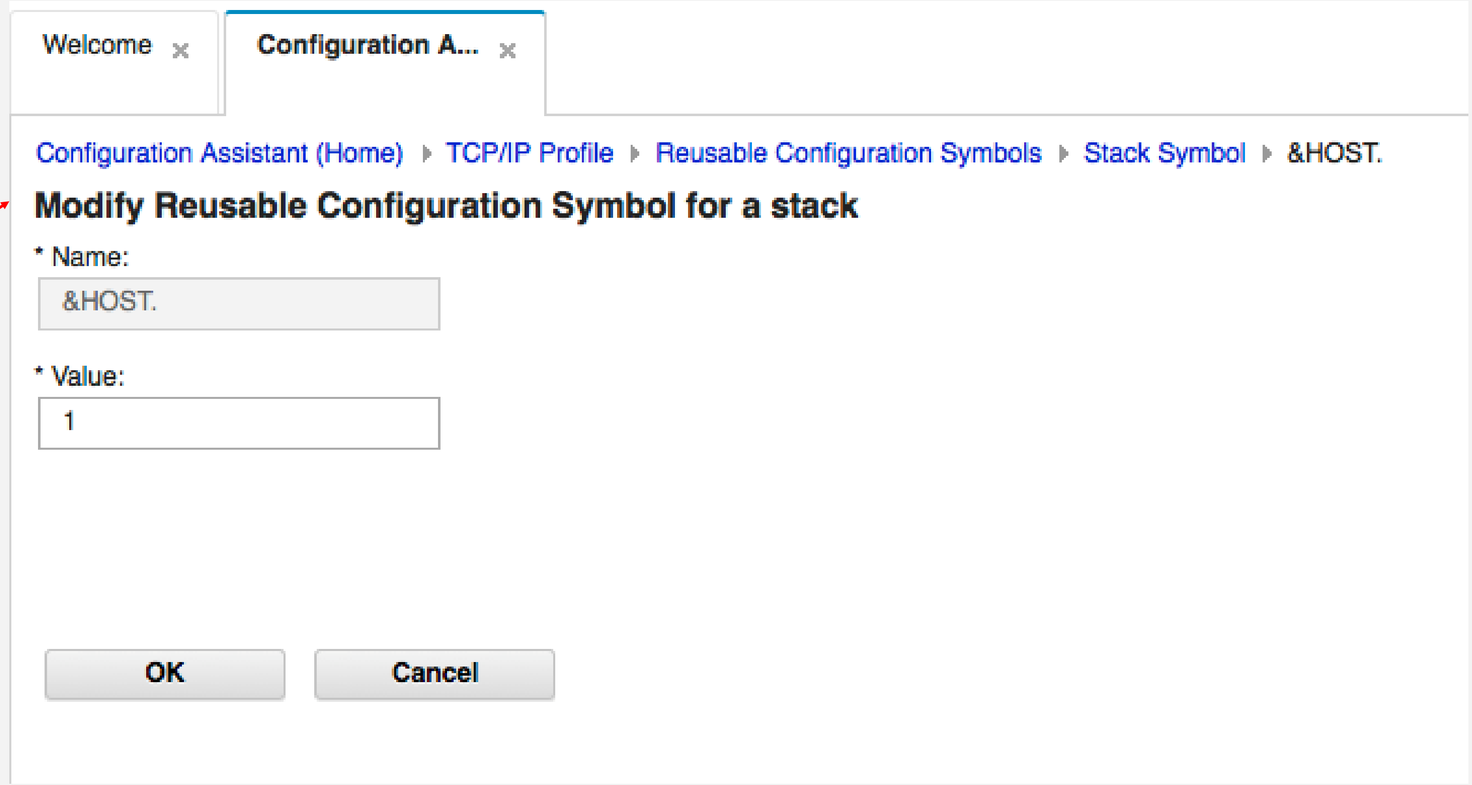


Next page

- This slide shows an example of setting a symbol through the reusable configuration that defines/ uses it.
- This interface gives you easy access to all stacks that need a value defined for this symbol.

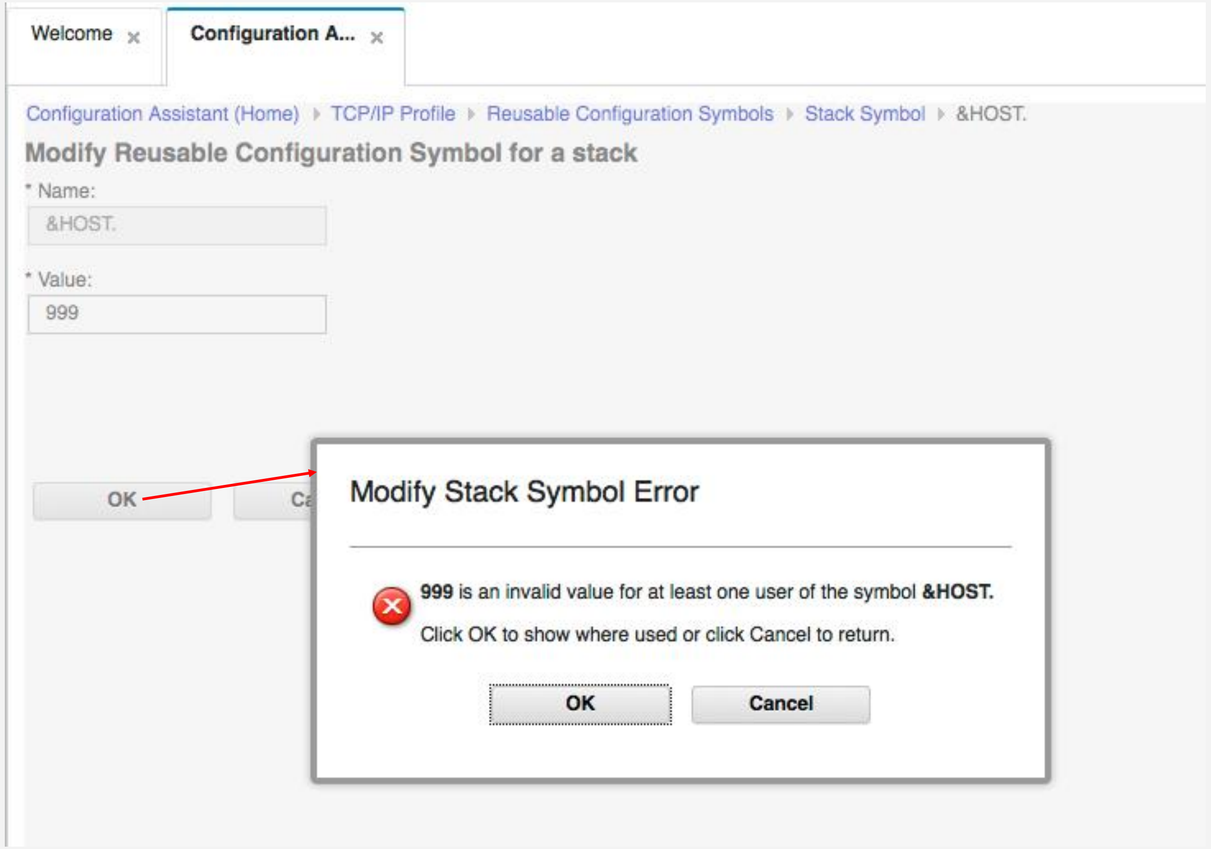
Resolving symbol values: via the RC, continued





- This slide shows an example of setting a symbol through the reusable configuration that defines/ uses it.
- This interface gives you easy access to all stacks that need a value defined for this symbol.

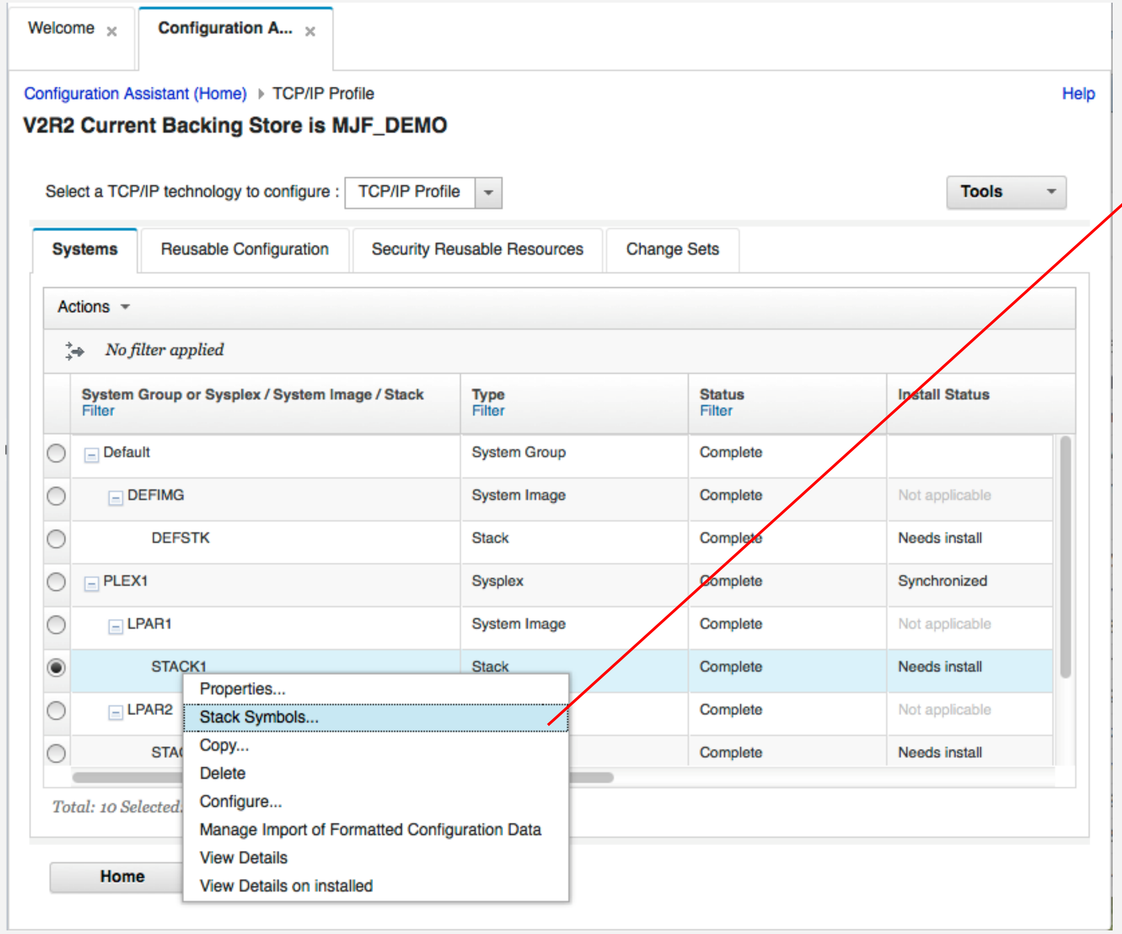
Symbols: automatic error checking



When you set a NCA stack symbol value, NCA verifies that the value you select won't cause any errors. This helps cut down on configuration errors!

- In this example, the user attempted to set &HOST. to 999.
  - Because &HOST. is used as an IP address octet, it cannot accept this value. Clicking OK on this error dialog shows all users of the symbol, to help you debug this error.

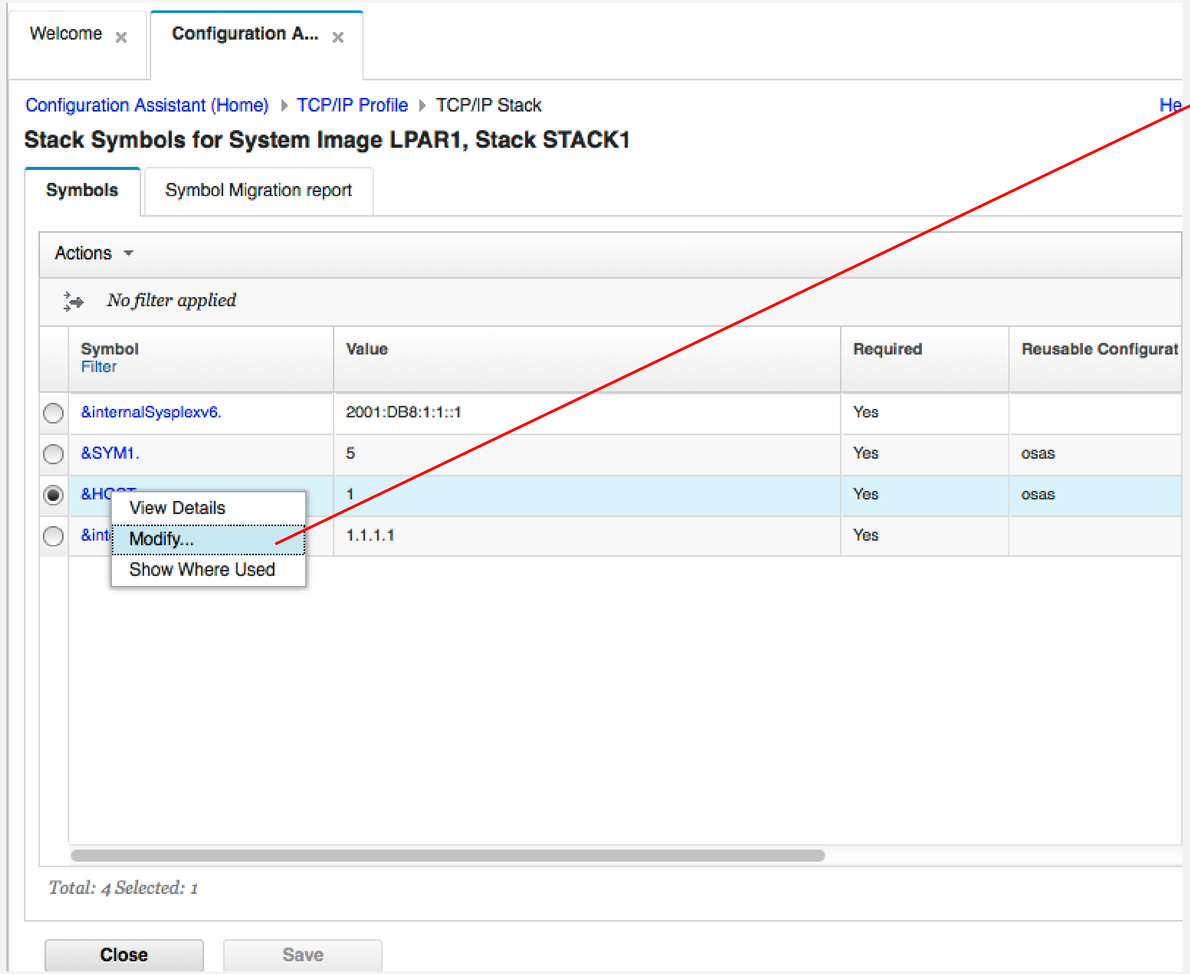
Resolving symbol values: via the stack

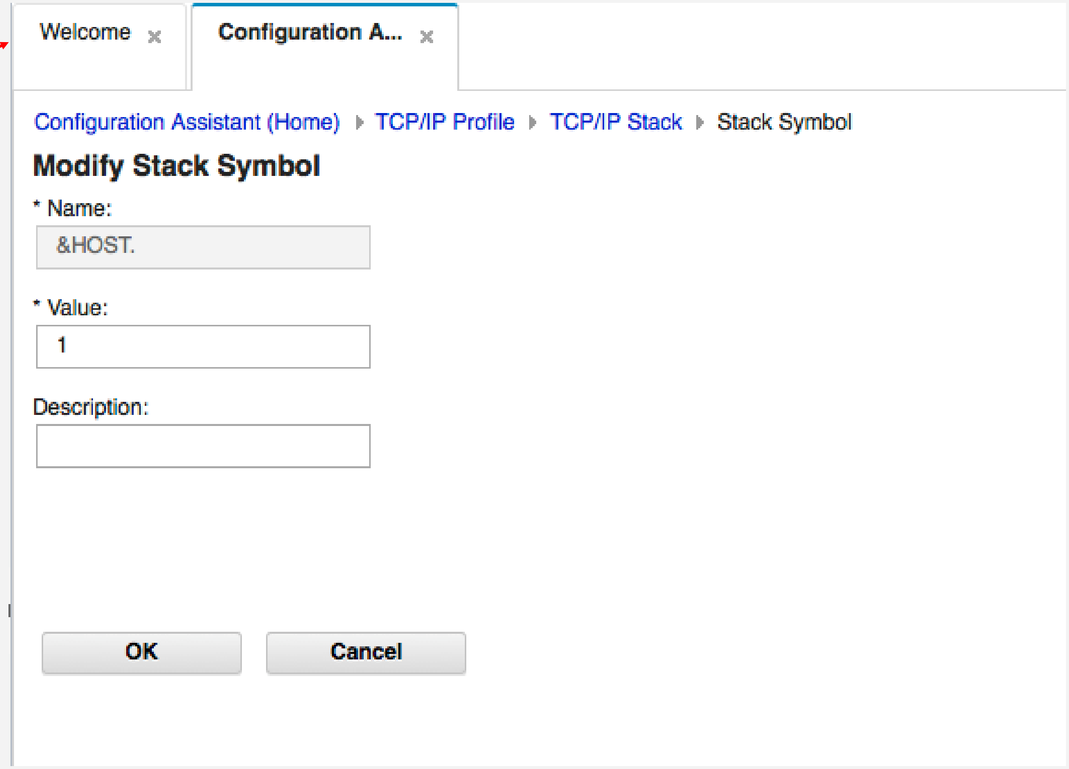


Next page

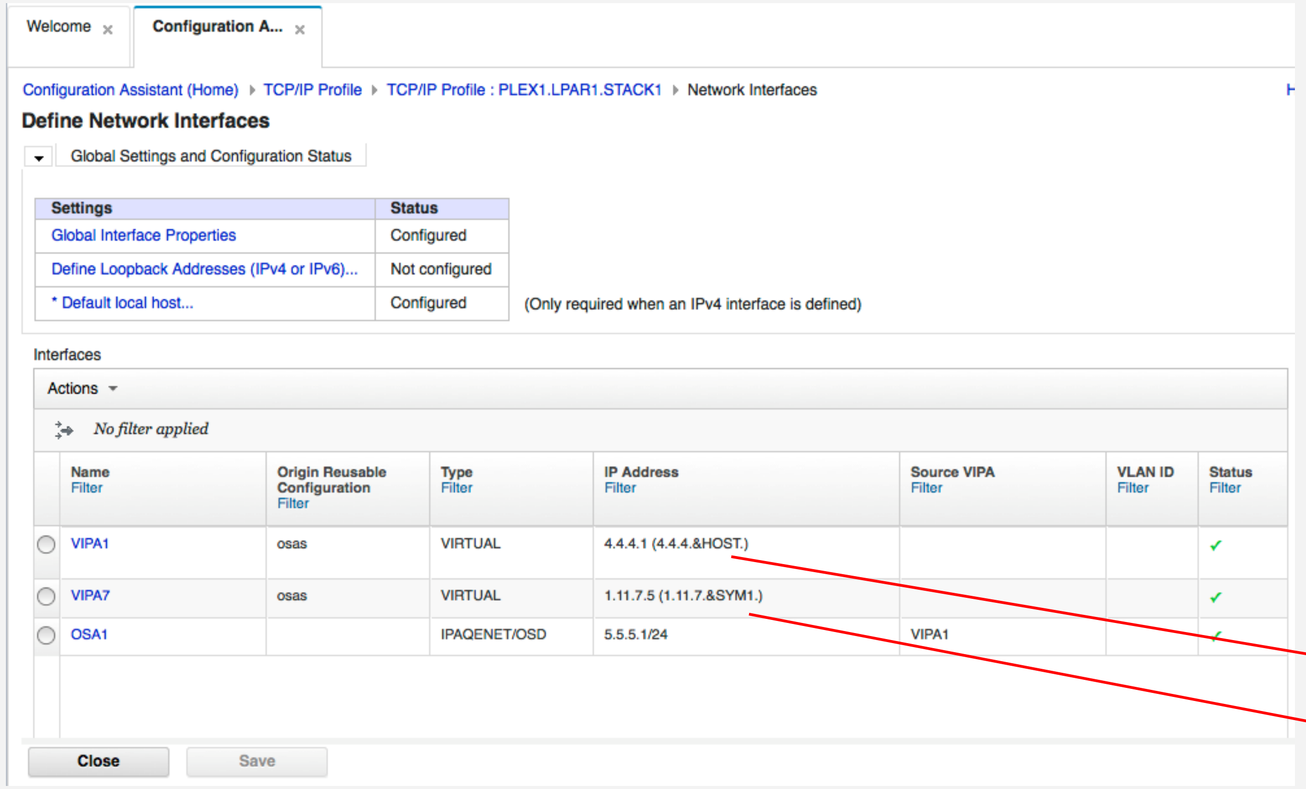
- This example shows how to resolve and update symbol values for a stack
- This interface gives access to all symbols used by a stack, across multiple reusable configurations

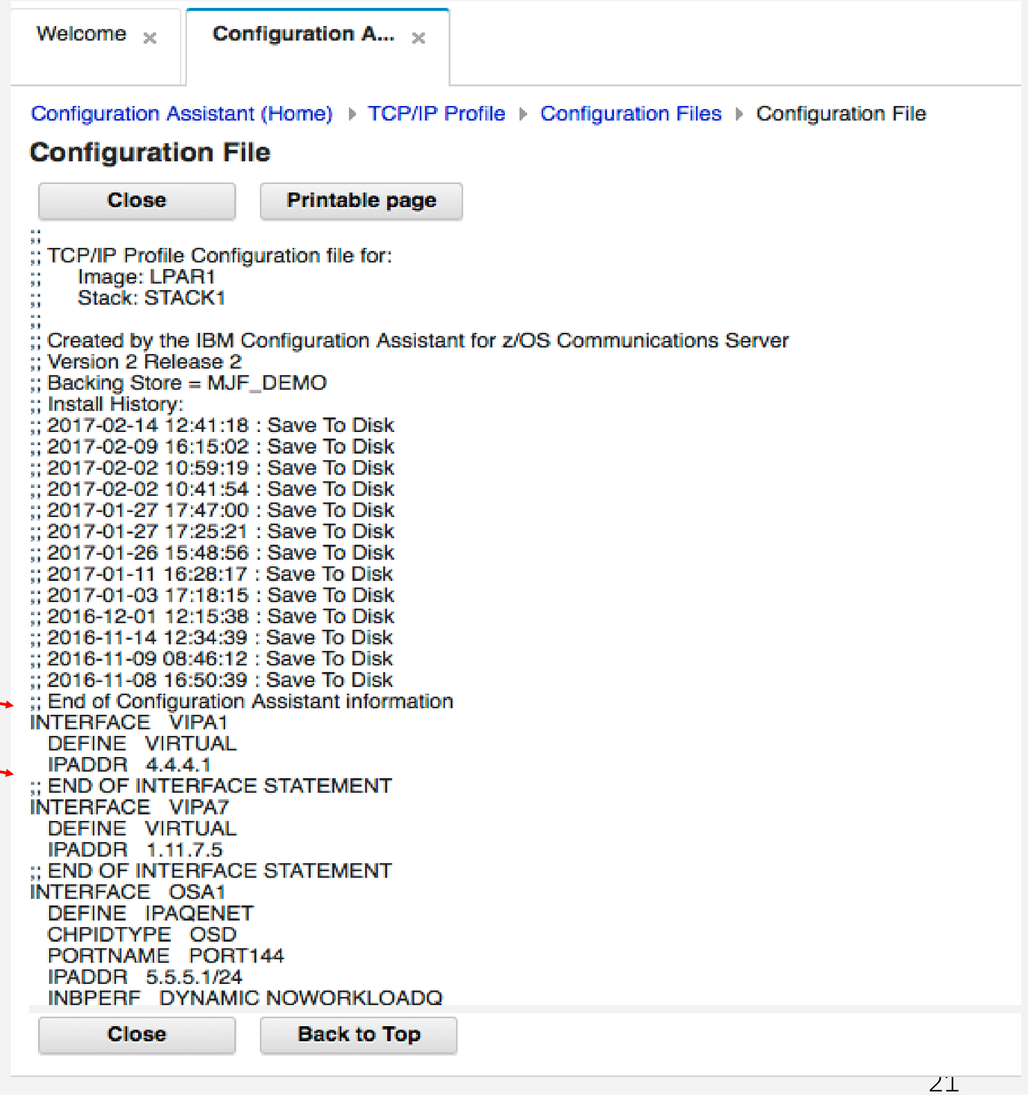
Resolving symbol values: via the stack, continued





Resolved symbols: generated configuration





- Network Configuration Assistant generates a TCP/IP profile with all NCA system symbols resolved.

IBM **z**