Push Registration

Overview

The objective is to allow a client node to register in the following cases:

1. Server is behind a firewall, so client cannot pull from it.
2. Group links are server and client push to each other.
3. Group links are server initiates all communication.

How Registration Currently Works

Client: On its scheduled interval, the PullService asks the RegistrationService to register with the server if it hasn't already done so. It goes into a loop that calls the DataLoaderService to pull from the server, then checks if it received its node identity.

Server: The PullUriHandler receives the pull request and looks up the node_security record for the requesting node. If the node_security record has registration_enabled flag on, then it asks the RegistrationService to register the node. It calls DataExtractorService to write a configuration batch to the stream.

GUI: The Manage → Nodes screen uses the registration_request table of status RQ to display registration attempts that need to be accepted. When selecting menu option Registration → Allow, it turns the registration_enabled flag on.

Database:
1. Client pulls registration URL, and server puts entry in sym_registration_request with status RQ.
2. User accepts the request, and server puts entries in sym_node with sync_enabled = 0, sym_node_host, and sym_node_security with registration_enabled = 1.
3. Client pull registration URL again, and receives configuration batch. Server updates sym_registration_request with status OK, sym_node with sync_enabled = 1, and sym_node_security with registration_time.
4. Client acknowledges configuration batch. Server updates sym_node_security with registration_enabled = 0.

Proposed Changes

The server will push the configuration batch to the client, and register the node if the batch is successfully loaded.

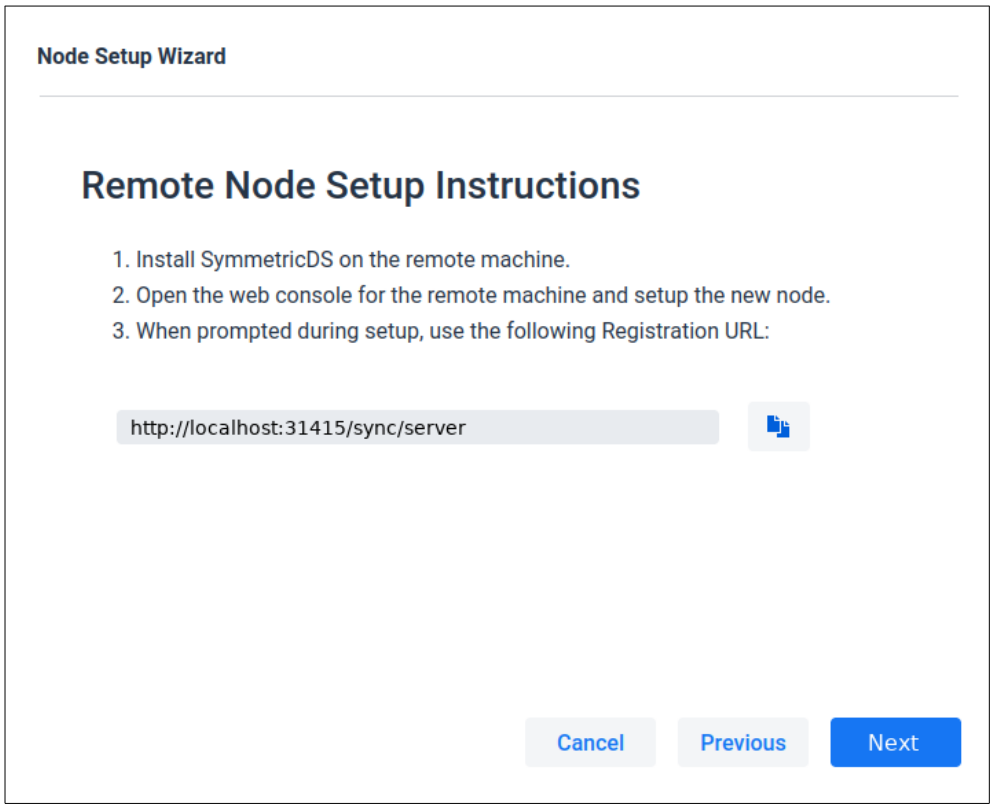
Add Node Wizard

From Manage → Nodes screen, after clicking "Add" button and selecting "Remote Node".

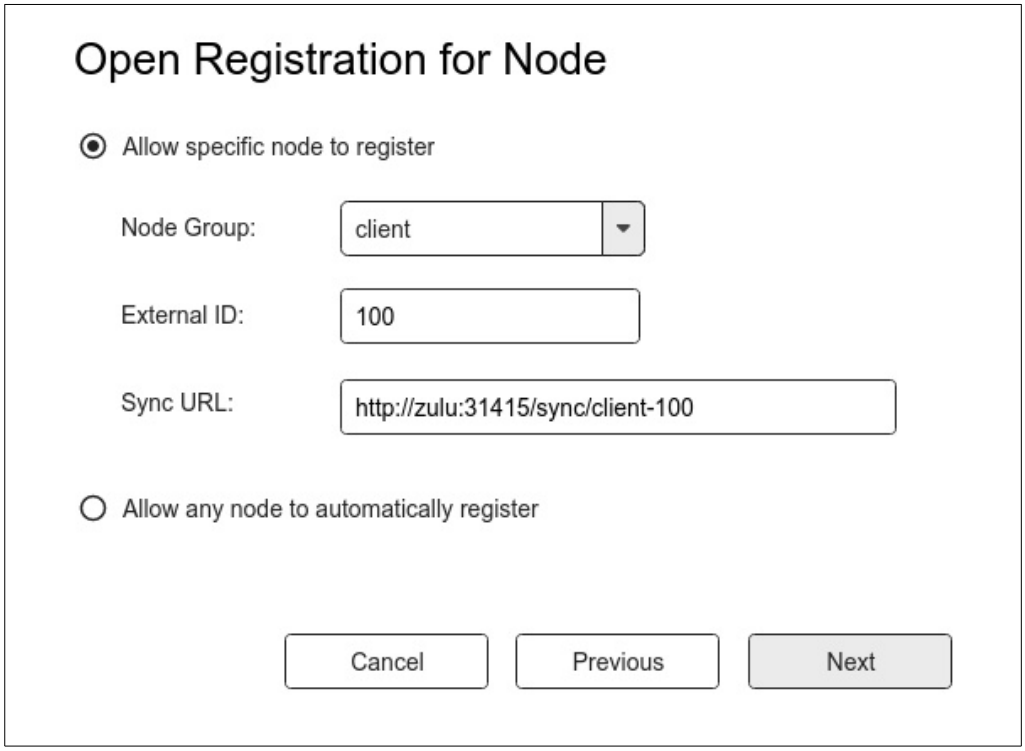
Modify the screen below with the following changes:

- Put the registration URL in a text box.
- Add a button with a "copy" icon that copies to clipboard from the text box. See https://cookbook.vaadin.com/copy-to-clipboard
- Remove the button "Turn On Auto Registration", with the functionality being moved to a subsequent screen in the wizard

- Change the “Finish” button to be labeled as “Next”



After clicking “Next”, the user can either enter a specific node to open registration for, or they can set the auto.registration parameter. If the auto.registration parameter is true, then that radio button is automatically selected. If the option for entering a specific node is used, then the wizard will disable auto.registration parameter, if needed, when finishing. If the option for allowing any node is used, then the “Next” button becomes “Finish”.

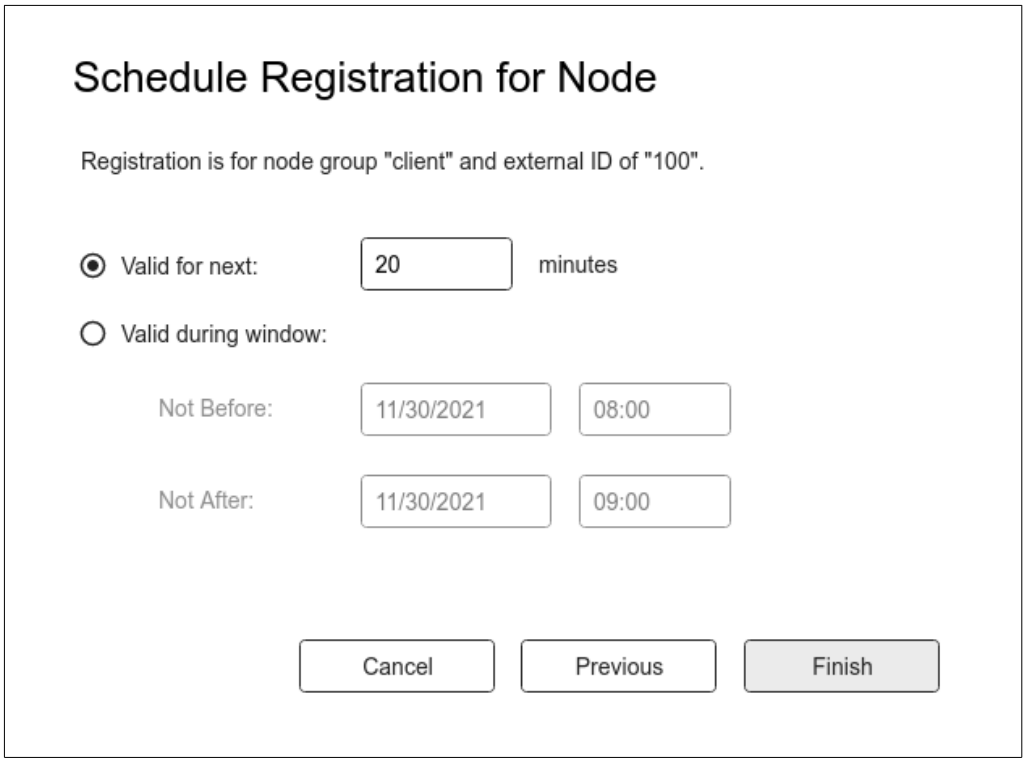


The selected “Node Group” in the drop-down determines if the “Sync URL” field is enabled. If the current node group is linked as a push to the selected node group, then “Sync URL” field is enabled. (Use ConfigurationService and call getNodeGoupLinkFor(parameterService.getNodeGroupId(), targetNodeGroupId, false).) The list of node groups for the drop-down can be retrieved as follows:

- Use NodeService and call findIdentity() to get current Node
- Create a Set<String>
- Use ConfigurationService and call getNodeGroupLinks(true), then loop for each
  - If source group ID equals current node group ID, add to the set
  - Else if target node group ID equals current node group ID, add to the set

The “External ID” field should be bound with the ExternalIdValidator. The “Sync URL” should be validated by constructing a URL object with it.

If clicking “Finish”, the auto.registration parameter is enabled, if not already enabled. If clicking “Next”, the user can schedule registration.



The “Not Before” and “Not After” fields could have icons and a date picker, similar to the AdvancedFilterWindow.

When clicking “Finish”, use Registration Service and call openRegistration(nodeGroup, externalId, syncUrl, notBefore, notAfter).

Node Registration Panel

If status contains “Connection refused” or “Connection timed out”, then display the help section for push registration.

Communication Service

Modify to return additional nodes that have both sync_enabled as false and registration_enabled as true. This will allow the Push Service to execute for remote nodes with open registration using a separate thread for communication.

Push Service

On the local node’s scheduled interval, the PushService will ask the RegistrationService to register any remote node if ALL of the following are true:

- Remote node has registration_enabled as true
- Group link from local to remote is push
- Local’s node_id is equal to remote’s created_at_node_id

Registration Service

Add a new registerToClient() method, called by Push Service, that does the following:

- Check that there is not an existing sym_registration_request to avoid race with pull
- Access registration URL of client with parameters nodeGroupId, externalId, hostName, and syncUrl
- Check HTTP code for WebConstants.SC_OK, and log any error
- Read URL encoded parameters in response and call processRegistration() as if it came from Registration URI Handler on a pull
- If node returned isSyncEnabled(), then access registration URL of client to send configuration batch

Registration URI Handler

Push requests are only handled if the node is unregistered, parameter registration.allow.push=true, and there is not an active pull registration that is getting HTTP codes returned.

1) If pushRegistration=true and method is POST:

- Verify syncUrl parameter matches the client's registrationUrl parameter
  - Return WebConstants.SC_FORBIDDEN
- Return URL encoded parameters normally sent in a registration request

2) If pushRegistration=true and method is PUT:

- Verify syncUrl parameter matches the client's registrationUrl parameter
  - Return WebConstants.SC_FORBIDDEN
- If all checks pass, load batch