

white paper

Easy Tag-Based Graphics

by *Stephen Holicky, Product Owner, Tridium*

Within the Niagara Community, we have long recognized the tagging challenge and its relationship role in enabling a more robust and intuitive UI. Before we can get to a future where ambient and voice-based computing, coupled with augmented reality informs the manner in which occupants interact with the built space around them, there is a demonstrable need to apply semantic tags to the universe of physical and digital things that are both inherently understood by humans and machines.

Defining a tagging strategy is now a prerequisite step for both new construction and retrofit projects which include building controls upgrades. Standard metadata tags enable software analysis of large quantities of equipment data and automated, data-driven monitoring and management of traditionally disparate building systems. They are a means of future-proofing a building automation system and accommodating for unforeseen data sharing needs with applications which may not have been invented yet. To ensure the standardization of data tagging in the 21st century, the building automation industry has written tagging requirements into Division 25

design specifications along with the recently proposed ASHRAE Standard 223.P.

With the most recent versions of Niagara 4, users can not only leverage tags to optimize

With each new release of Niagara Framework® Tridium has evolved its tagging support to enable easier, automated and more flexible methods of applying and using semantic tags. In the upcoming 4.10 release, we will be introducing a new browser friendly HTML5 tagging editor which will significantly improve your workflow and user experience.

their operations but can also use tags as the basis for graphical visualizations. This whitepaper presents a practical application of how to efficiently incorporate tags into Niagara station visualizations in the most foolproof way.

TAG-BASED PX GRAPHICS

As of Release 4.6, Niagara Framework has supported responsive, mobile-friendly Px graphics which conforms the UI to fit within various screen sizes. With Release 4.9, Tridium developers have added tag-based PX bindings, and now in the upcoming Release 4.10, we will be introducing a new HTML5

tagging editor. Rather than using bespoke point naming conventions to resolve and render your associated graphical components when designing a customized UI, you can apply and then leverage point tagging for your station's visualizations. The tags utilized by tag-based bindings can be pulled from multiple custom tag dictionaries, as well as from the Project Haystack standard. Since the Niagara Framework is tag-agnostic, this new feature provides the utmost flexibility when applying tags to Niagara stations and building systems. The result is visualizations that are reusable and manufacturer- and equipment-agnostic. Tag-based bindings streamline on-site efficiency, save time and provide more flexibility when deploying graphic templates on new and existing Niagara stations.

HOW IT WORKS

Tag-based Px bindings utilize the Niagara Entity Query Language, or NEQL, within a standard object resolution descriptor (ORD) to resolve graphical components. NEQL-based ORDs are resolved on the station-side instead of on the client-side (or browser) and are not executed across the entire station, but rather scoped to descendants of, or entities related-to, the component on which the graphic view is placed. Once the graphic is loaded, the underlying NEQL queries are not resolved again to ensure consistent UI responsiveness.

Before you can implement tag-based Px bindings in a station, a tag dictionary needs to be installed, followed by the tagging of its components and specification of all necessary

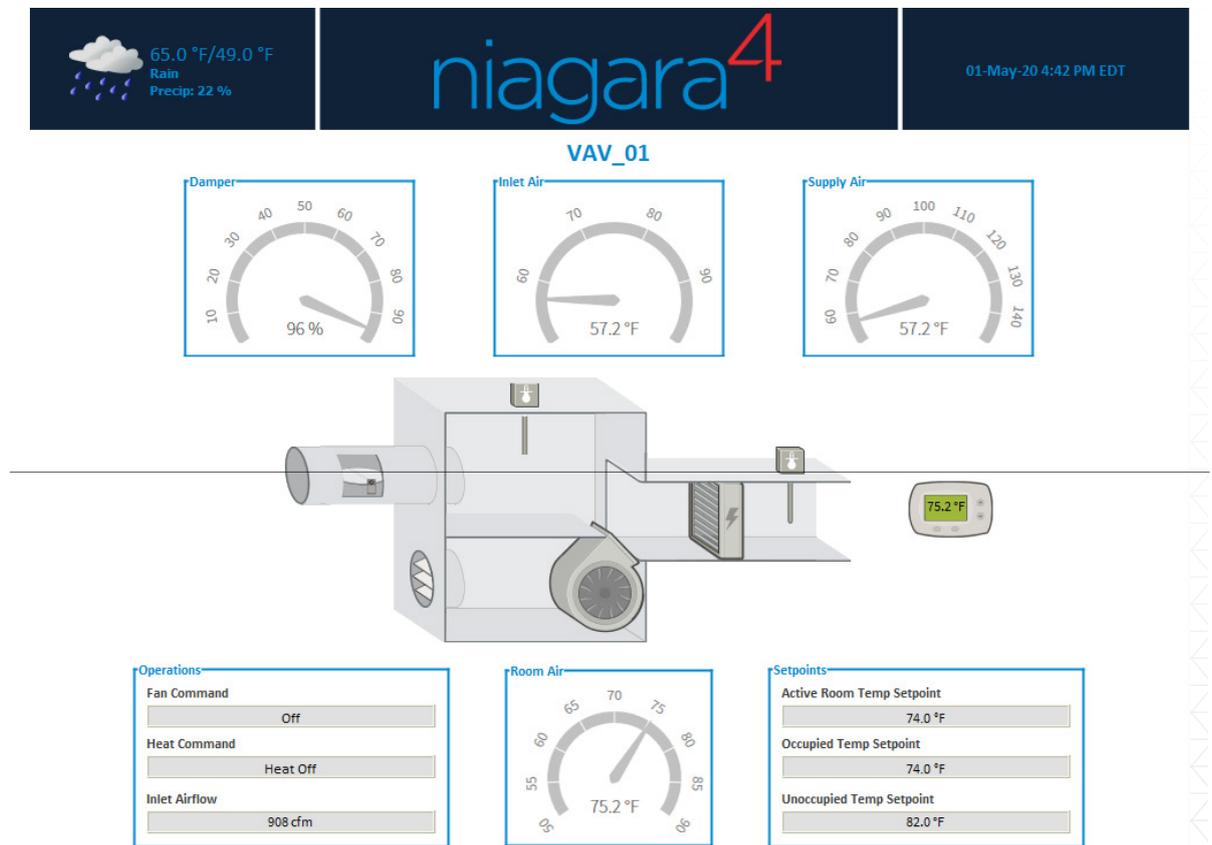


FIGURE 1. AHU VAV GRAPHIC

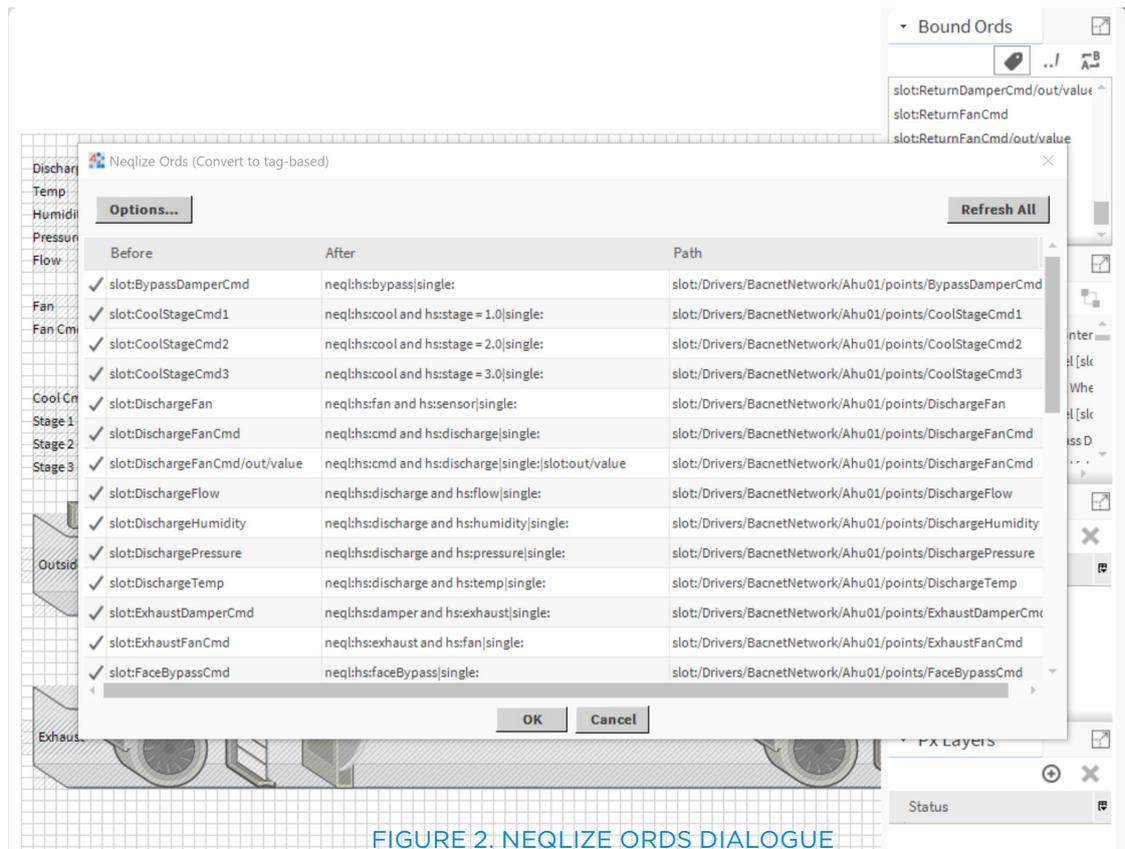


FIGURE 2. NEQLIZE ORDS DIALOGUE

entity relationships. To create the example VAV graphic shown in Figure 1, the Niagara and Haystack tag dictionaries, as well as a custom tag dictionary were installed. Here, the custom tag dictionary had specific tag rules which automatically applied Haystack tags to the station components based on defined conditions.

Creating a tag-based Px binding is a matter of choosing from a cascade of options available from the Px Editor:

- ▶ Neqlize Ords automatically converts direct or relative ORDs to a NEQL based ORD and will save considerable time when incorporating tags in your Niagara visualizations (Figure 2).
- ▶ Neqlize Options launches a dialogue for specifying the conversion mode. From this menu, you can also exclude specified relations and tags that were utilized for the Neqlize Ords feature (Figure 3).
- ▶ Neqlize Options - Conversion Mode prompts you to select between Traverse If Possible, Traverse Only or Select Only. A Select statement is a NEQL statement for selecting a collection

of Entities, whereas a Traverse statement is a NEQL statement for selecting a collection of Entities by traversing a relation from either end of the relation. The default setting is “Traverse if Possible”, which ensures both statements will be utilized upon its conversion.

- ▶ Neqlize Options - Excluded Tags and Relations: You may want to exclude specific tags and relations from the conversion process. In this example, certain tags and relations that are commonly used within a Niagara station have been excluded. Including an n:child relationship, for example, would yield multiple results during the conversion process, thereby needlessly complicating or invalidating the final NEQL ORD. So this tag has been excluded.

PRACTICAL APPLICATION

How would tag-based bindings be of practical use in some common building control situations? Consider the case of an air handling unit, or AHU, that is served by two different field controllers from different OEMs. Ideally, you would want to redeploy a preexisting AHU graphic for this unit without having to reconfigure its ORDs.

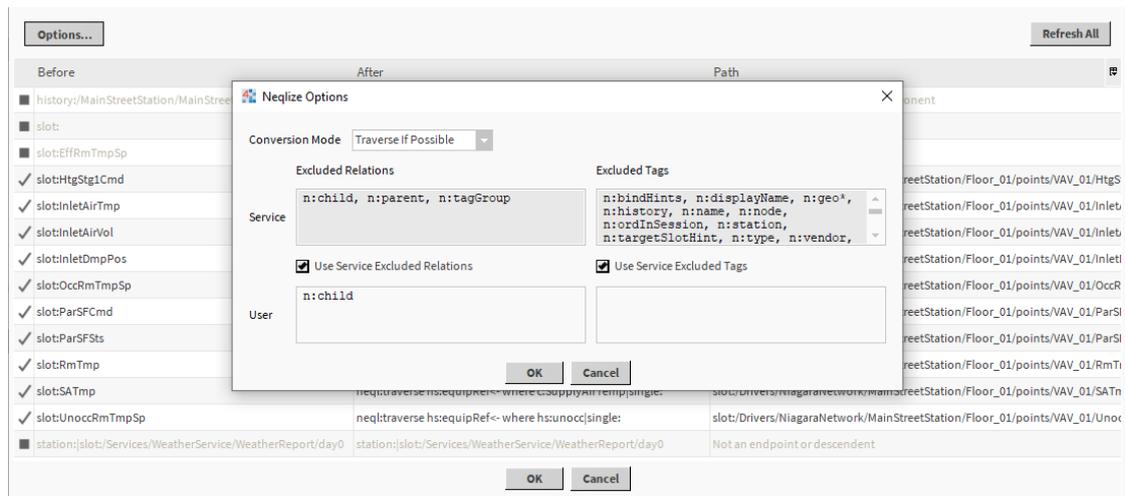


FIGURE 3. NEQLIZE OPTIONS DIALOGUE

Prior to Niagara 4.9 and tag-based Px Bindings, you would have had to edit each ORD according to the component architecture of the original graphic view. This exercise can take up valuable time when trying to get the job done right. Even worse, if the component names are non-intuitive or even remotely complicated, it can be an exhaustive effort to ensure each ORD matches the component architecture such that the graphic points resolve correctly. Additionally, you would be unable to utilize a relativized graphic for this scenario. The only option would be to manually edit the ORDs. As of Niagara 4.9 and tag-based bindings, the process is simplified, as all you need to do is NEQLize the graphic ORDs and then define it as a view on a component.

For the AHU under discussion, you would simply need to convert the ORDs within the existing AHU graphic using Neqlize Ords and

then define it as a view on a device folder named AHU02. It is now this easy to reuse graphics in a Niagara station or across your enterprise portfolio of facilities.

SUMMARY

Tag-based Px Binding is another step toward easier customization of UI graphics for Niagara users, thus resulting in a better UI experience for end-customers. This streamlined way to convert-slot path ORDs to tag-based NEQL ORDs improves the reusability of a Px view and saves time in the process. Tridium is striving to make effective tagging of assets and data easier with each release of Niagara. You can find additional information on tagging, relations and NEQL by searching in the Resource Center at www.niagara-community.

About the Author:

Stephen Holicky is a Niagara Product Manager with experience spanning building systems engineering, smart building master planning/design, specification development, commissioning and energy auditing.

TRIDIUM
tridium.com
Headquarters
North America
1 804 747 4771

Support
North America & Latin America
1 877 305 1745

Europe, Middle East & Africa
44 1403 740290

Asia Pacific
8610 5669 7148

© 2020 Tridium Inc. All rights reserved. All other trademarks and registered trademarks are properties of their respective owners.

Information and/or specifications published here are current as of the date of publication of this document. Tridium, Inc. reserves the right to change or modify specifications without prior notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia. Products or features contained herein may be covered by one or more U.S. or foreign patents. This document may be copied only as expressly authorized by Tridium in writing. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form.