Do scripts work the same in New Exalate and Classic?

Last Modified on 02/09/2026 3:05 am EST

Yes, the core scripting engine and Groovy-based sync logic remain the same between Classic and New Exalate. Your existing outgoing and incoming sync scripts use the same replica concept, helper functions, and scripting patterns in both versions. This consistency means teams familiar with Classic scripting can transition to the new experience without relearning synchronization logic.

The fundamental approach stays identical, where outgoing scripts populate the replica object with data to share and incoming scripts apply that data to local work items. Helper functions like commentHelper.mergeComments, attachmentHelper.mergeAttachments, and nodeHelper methods work the same way across both experiences.

What changes is how you interact with these scripts through the interface. The new experience presents sync rules in a unified side-by-side view rather than separate consoles for each platform. You also gain access to Aida for AI-assisted script generation and troubleshooting, versioning to track changes over time, and Test Run to validate script behavior before deployment.

When importing existing connections from Classic to the new experience, your sync scripts transfer directly without requiring modifications. The platform maintains backward compatibility with Classic scripting patterns while offering enhanced tools for script development and testing in the new interface. Organizations can leverage existing scripting knowledge and documentation when working in either environment.