WHITE PAPER / VERIFIED MILESTONE

Native applications, one distributed fabric.

Hypha Windows Workspace acceptance | 28 October 2026

Executive summary

Hypha reached a meaningful architectural milestone: a person selected Notepad inside a browser-based Hypha Workspace, and an authorized job opened the native Microsoft Windows application on the designated EVO-T1 computer. Workspace reported success, and the Windows process identity was independently verified.

This marks the shift from Hypha installed on Windows and Linux toward native applications launched through Hypha. The operating systems retain control of their own desktops.

ACCEPTED WINDOWS CANARY

Notepad

Microsoft.WindowsNotepad | EVO-T1 | Session 1

SUCCEEDED

Why it matters

- **One user experience**

  Applications can be presented as ordinary Workspace choices, without exposing orchestration details.

- **Native OS cooperation**

  Hypha works with Windows activation and process identity rather than bypassing them.

- **Security stays explicit**

  Human sessions, host targeting, signed catalogs, and process attestation remain in the execution path.

- **A cross-platform foundation**

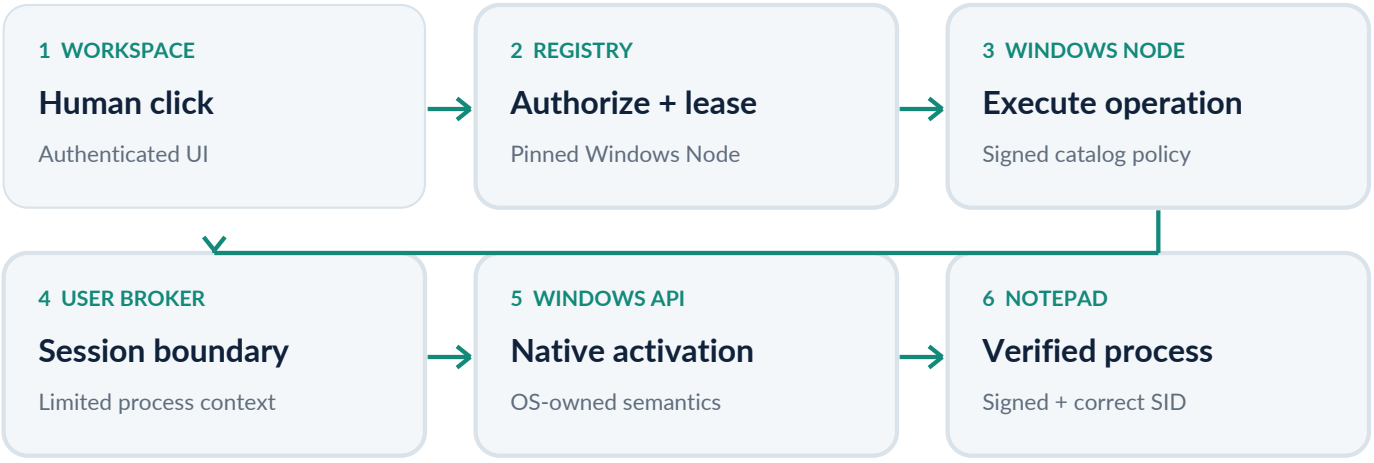
  Linux desktop work and this Windows launch support the next step: one application catalog.

Scope: This document records an accepted Windows launch and prior Linux desktop integration progress. It does not claim that every application is supported, that remote desktop streaming is implemented, or that independent scale benchmarks have been completed.

HOW THE SYSTEM WORKS

The native application path

The browser asks Hypha to perform a named operation. Hypha authorizes the request, routes it to a pinned host, and uses Windows-native mechanisms to activate and attest the application.



Security and trust boundaries

Human authorization. Workspace uses a distinct human session. The Registry enforces durable grants on requests; a visible tile is not itself permission to run an operation.

Controlled target and executable. The Notepad tile pins the accepted Windows node and submits the cataloged **windows.desktop.launch.v1** operation for **Microsoft.WindowsNotepad**. It does not accept arbitrary executable paths.

Windows-native verification. The broker checks the running user/session, installed package path, valid Microsoft signature, and process ancestry. Ambiguous or mismatched candidates fail closed.

Key implementation lesson: On EVO, Windows activation returned a short-lived parent PID while Notepad continued as its child. The accepted repair verified that relationship and the child's creation time rather than assuming the two PIDs must be equal.

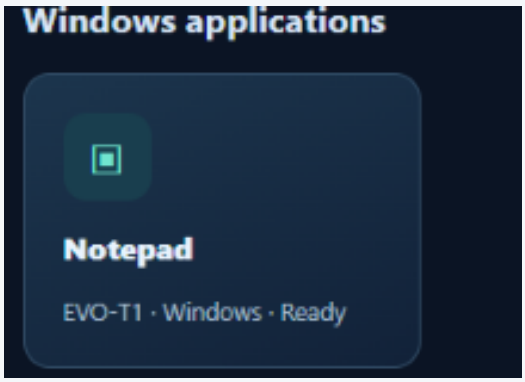
OBSERVED RESULTS

From a browser click to Notepad

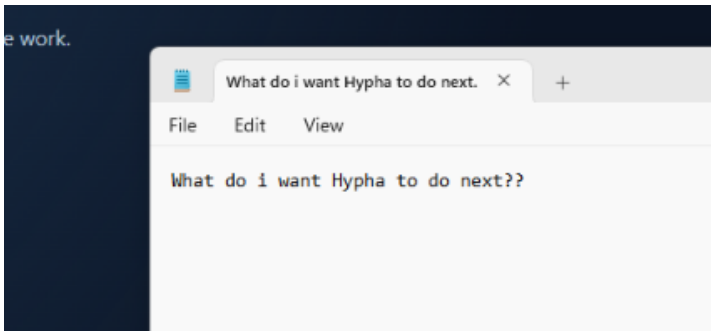
On October 28, 2026, the Registry and Windows evidence closed two independent acceptance gates: a signed native launch and then the end-user Workspace click.

Native Windows job	c440b852-4621-48a9-b61c-5855f0e837d0
Native process	Notepad.exe / PID 10800 / Session 1
Workspace click job	21ecf2a8-5858-43aa-89eb-b0b1a5eebebb
Workspace target	EVO-T1 / Windows / Ready
Job outcome	ok: true / native Notepad visible

WORKSPACE TILE



NATIVE WINDOWS APPLICATION



Opened on EVO-T1; job status: succeeded.

Figure 1. User-provided screenshots: Notepad Ready in Workspace and visible on Windows.

Independent OS attestation: the successful native canary reported the signed Microsoft executable, correct user SID, Session 1 and expected package path. Node and accelerator processes were left untouched; no new broker stderr lines were observed.

MILESTONE, NOT A FINISH LINE

What this proves - and what comes next

PROVEN

An authenticated Workspace click can launch a real, signed Windows application through an authorized Hypha job, targeted to EVO and verified in its interactive session.

NOT YET PROVEN

Universal app compatibility, cross-device app streaming, repeated Notepad tab behavior, native macOS support, or published performance and scale benchmarks.

Practical next steps

01 **Linux Workspace canary**
Expose and accept an existing Linux native desktop operation from the same user-facing Workspace.

02 **Catalog-driven application UX**
Replace the single-machine Notepad binding with approved app discovery, host selection, and readiness checks.

03 **Windows lifecycle refinements**
Validate already-running Notepad and new-tab behavior without weakening process lineage and session verification.

04 **Broader platform acceptance**
Advance Windows app coverage and validate macOS-native application support only after implementation.

Reproducibility and project record

Windows Node source: commit **8cf2a951** (96 tests passed).
Live Workspace frontend: commit **f6fa1ded**; SHA256 begins **47470154...**.
Final Registry checkpoint: **acb5785819232230190eb900646617e0fed67f13**, pushed to **workspace-windows-notepad-20261028**; five training-ready examples saved.

Notes: The Registry feature branch has a recorded checkpoint, not a confirmed merge to main.

HYPHA | Patent Pending

© 2026 TKT, Inc. All Rights Reserved.

HYPHA technology is the subject of pending patent applications. The contents of this white paper, including its text, illustrations, diagrams, and technical descriptions, are protected by applicable copyright laws. No portion may be reproduced, distributed, or transmitted without prior written permission from TKT, Inc.