

VERSIONHUB

SOFTWARE CENTRAL VERÖFFENTLICHEN UND AKTUELL HALTEN



MASTER - ORANGE



CLIENT - BLAU

MASTER	CLIENT
Versionen als geprüfte Pakete veröffentlichen	Lokalen Stand prüfen und Updates installieren
NAS ODER CLOUD-SYNC	SICHERHEIT
Gemeinsamer Ordner ohne zusätzlichen Serverdienst	SHA-256, Sicherung und atomarer Katalog

Version 1.2.2 | Windows 10/11 x64

1. Überblick

- VersionHub verteilt selbst erstellte Windows-Programme zentral über einen gemeinsamen Ordner. Der Master veröffentlicht Versionen, der Client zeigt den lokalen Stand und installiert freigegebene Updates.
- Als Repository eignet sich ein NAS-Netzlaufwerk oder ein lokal synchronisierter Cloud-Ordner. VersionHub selbst benötigt keinen Serverdienst und überträgt keine Daten an einen Herstellerdienst.

2. Bestandteile

- VersionHub.Master.exe: pflegt den zentralen Katalog und veröffentlicht unveränderliche ZIP-Pakete. Das orange-goldene Icon kennzeichnet die Verwaltungsanwendung.
- VersionHub.Client.exe: ordnet Apps einem PC zu, vergleicht Versionen und bereitet Updates vor. Das blau-cyane Icon kennzeichnet die Client-Anwendung.
- VersionHub.Updater.exe: installiert das vorbereitete Paket in einem eigenen Prozess und verwendet ebenfalls das blaue Client-Icon. Der Client bleibt dabei geöffnet.

3. Repository vorbereiten

- Legen Sie auf dem NAS oder im synchronisierten Cloud-Bereich einen eigenen Ordner an, zum Beispiel \\NAS\Software\VersionHub.
- Der Master-Benutzer benötigt Lese- und Schreibzugriff. Client-Benutzer sollten nur Leserechte erhalten. Diese Trennung schützt Katalog und Pakete vor unbeabsichtigten Änderungen.
- Bei Cloud-Nutzung muss der Ordner auf jedem PC lokal synchronisiert und vollständig verfügbar sein. Tragen Sie im Client den jeweiligen lokalen Sync-Pfad ein.

4. Eine Version veröffentlichen

- Starten Sie den Master und wählen Sie den Repository-Ordner.
- Vergeben Sie eine dauerhaft gleichbleibende App-ID, zum Beispiel meinefirma.meineapp. Geben Sie Anzeigename und eine aufsteigende Version wie 0.01, 0.01.1 oder 1.2.0 an.
- Wählen Sie als Installationsart Programmordner, MSI-Paket oder Setup-EXE. Der Quellordner enthält immer alle für diese Version benötigten Dateien.
- Beim Programmordner wird die optionale Startdatei relativ zum Quellordner angegeben, zum Beispiel MeineApp.exe.
- Mit VERSION VERÖFFENTLICHEN erstellt VersionHub ein ZIP-Paket, berechnet SHA-256 und aktualisiert danach atomar den Katalog. Eine bereits veröffentlichte Version wird nicht überschrieben.

5. Vorhandene App als neue Version

- Markieren Sie die vorhandene App in der Tabelle und klicken Sie AUSWAHL ÜBERNEHMEN. Ein Doppelklick auf die Tabellenzeile führt denselben Schritt aus.
- Der Master übernimmt App-ID, Anzeigename, Installationsart, Start- oder Setupdatei, Setup-Argumente, Erfolgscode, Versionsdatei und Optionen. Die App-ID wird gesperrt, damit die neue Version sicher derselben App zugeordnet bleibt.
- Die nächste Versionsnummer wird vorgeschlagen: Aus 0.01 wird 0.02, aus 0.01.009 wird 0.01.010, aus 1.2.9 wird 1.2.10 und aus 2.0.0-beta wird 2.0.0. Führende Nullen bleiben erhalten. Der Vorschlag kann vor der Veröffentlichung geändert oder um weitere Zahlenblöcke ergänzt werden, zum Beispiel von 0.01 auf 0.01.1.
- Versionsnummern dürfen aus beliebig vielen durch Punkte getrennten Zahlenblöcken bestehen. Jeder Block wird numerisch verglichen; deshalb ist 0.01.1 höher als 0.01. Vorabkennzeichnungen wie -beta sind weiterhin möglich.
- Nach einer Veröffentlichung merkt sich der Master den zuletzt verwendeten Quellordner für diese App und dieses Repository. Bei bereits früher angelegten Apps muss der aktuelle Quellordner einmal neu gewählt werden.
- Ändern oder bauen Sie das Programm im Quellordner, ergänzen Sie die neuen Hinweise und veröffentlichen Sie anschließend. NEUE APP leert das Formular und entsperrt die App-ID.

6. MSI oder Setup-EXE veröffentlichen

- Setupdatei: Wählen Sie die EXE- oder MSI-Datei innerhalb des Quellordners. Begleitdateien dürfen daneben in Unterordnern liegen und werden gemeinsam verpackt.
- Setup-Argumente: Für MSI ist /qn /norestart voreingestellt. Bei Setup-EXE gelten die Parameter des verwendeten Setup-Systems. Für ein sichtbares interaktives EXE-Setup kann das Feld leer bleiben.
- Erfolgscodes: Standard sind 0, 1641 und 3010. 1641 und 3010 bedeuten beim Windows Installer einen erfolgreichen Abschluss mit erforderlichem beziehungsweise bereits angestoßenem Neustart. VersionHub startet den PC nicht selbst neu.
- Versionsdatei: Optional, aber empfohlen. Geben Sie den vollständigen Pfad zur installierten Haupt-EXE an, zum Beispiel %ProgramFiles%\Firma\MeineApp\MeineApp.exe. Der Client liest daraus Produkt- oder Dateiversion.
- Setup als Administrator: Aktivieren, wenn das Setup systemweit installiert oder geschützte Ordner beziehungsweise Registry-Bereiche verändert.

7. Einen Client einrichten

- Kopieren Sie den vollständigen Client-Ordner auf den Ziel-PC. Client und Updater müssen nebeneinander liegen.
- Starten Sie den Client, wählen Sie dasselbe Repository und klicken Sie PRÜFEN.
- Wählen Sie eine App. Nur beim Programmordner geben Sie einen Installationsordner an; bei MSI und Setup-EXE bestimmt das Setup den Zielordner. Speichern Sie die Zuordnung.
- Klicken Sie AUSGEWÄHLTES UPDATE INSTALLIEREN. Das Paket wird lokal kopiert, per SHA-256 geprüft und entpackt. Der Client bleibt während der Installation geöffnet und prüft den Stand danach automatisch erneut. Falls konfiguriert, fordert Windows vor dem Setup Administratorrechte an.

8. Verhalten bei Updates

- Programmordner: Vor jeder Installation wird der vorhandene Zielordner unter %LOCALAPPDATA%\VersionHub\backups gesichert.
- Dateien aus dem neuen Paket werden ersetzt. Dateien, die zu einer älteren Paketversion gehörten und jetzt fehlen, werden entfernt. Andere Dateien, etwa lokale Benutzerdaten, bleiben erhalten.
- Scheitert der Kopiervorgang, stellt der Updater die Sicherung wieder her. Der lokale Versionsstand wird erst nach erfolgreicher Installation aktualisiert.
- MSI und Setup-EXE: VersionHub wartet auf das Setup und speichert die neue Version nur bei einem freigegebenen Rückgabecode. Ist eine Versionsdatei angegeben, muss diese danach vorhanden sein und mindestens die veröffentlichte Version melden.
- Bei Setup-Installationen ist das jeweilige Setup für Reparatur, Rücknahme und Sicherung verantwortlich. VersionHub verändert den vom Setup gewählten Programmordner nicht direkt.

9. Sicherheit und Betrieb

- Die SHA-256-Prüfung verhindert die Installation eines Pakets, das nicht zum Katalogeintrag passt. Sie ersetzt keine digitale Signatur: Wer Schreibzugriff auf Katalog und Pakete hat, könnte beide verändern.
- Beschränken Sie deshalb Schreibrechte auf den Master-PC beziehungsweise das Veröffentlichungskonto. Sichern Sie das Repository regelmäßig und verwenden Sie für sensible Umgebungen zusätzlich signierte Programme.
- MSI- und EXE-Pakete führen Code mit den Rechten des Updaters aus. Veröffentlichen Sie ausschließlich vertrauenswürdige, selbst erstellte oder digital signierte Setups und prüfen Sie Silent-Parameter zuerst auf einem Test-PC.

- VersionHub verwaltet keine Zugangsdaten und überträgt keine Telemetrie. Katalog und Pakete verbleiben im gewählten Repository.

10. Fehlerbehebung

- Repository nicht erreichbar: Netzwerkverbindung, Laufwerkszuordnung oder Cloud-Synchronisierung prüfen.
- Prüfsumme stimmt nicht: Paket nicht installieren; Repository erneut synchronisieren oder die Version im Master unter einer höheren Versionsnummer neu veröffentlichen.
- Keine Schreibrechte: einen beschreibbaren Installationsordner wählen oder die angezeigte Windows-Administratorfreigabe bestätigen.
- Setup-Rückgabecode nicht erfolgreich: Setup-Protokoll und Herstellerparameter prüfen. Den Code nur dann zur Erfolgsliste hinzufügen, wenn er laut Setup-Dokumentation tatsächlich Erfolg bedeutet.
- Versionsdatei nicht erkannt: Pfad, Umgebungsvariablen und die in der EXE hinterlegte Produktversion oder Dateiversion prüfen.
- Updater-Fehlerprotokolle liegen unter %LOCALAPPDATA%\VersionHub\logs. Sicherungen liegen unter %LOCALAPPDATA%\VersionHub\backups.

11. Neu in Version 1.2.2

- Master und Client besitzen jetzt getrennte, farblich eindeutig erkennbare Programmsymbole bei unverändertem VersionHub-Motiv.
- Orange und Gold stehen für den Master und die zentrale Veröffentlichung. Blau und Cyan stehen für Client und Updater auf den Ziel-PCs.
- Beide Windows-Icons enthalten optimierte Größen von 16 bis 256 Pixel und bleiben dadurch in Titelleiste, Explorer, Verknüpfung und Taskleiste klar erkennbar.
- Flexible Versionsnummern und der dauerhaft geöffnete Client aus Version 1.2.1 bleiben vollständig erhalten.

Hinweis: Testen Sie neue Versionen zuerst auf einem einzelnen Client und halten Sie eine unabhängige Datensicherung vor.