

TECHNICAL EVIDENCE REPORT

Delphi App Translation Studio

Group 0 Evidence and Ownership Report

Completed August 9, 2026

Group 0 complete - no remediation source code changed

Documentation refreshed: August 25, 2026

Table of Contents

Appendix A. License Information	8
1. Executive conclusion.....	3
2. Preservation and safety record.....	3
3. Reconciliation of 179 and 173 entries.....	4
3.1 Compared catalogs.....	4
3.2 Stable-key comparison.....	4
3.3 Source proof.....	4
3.4 Decision.....	5
4. TC-18 ownership analysis.....	5
4.1 Observed failure.....	5
4.2 Static execution trace.....	5
4.3 Pristine-versus-pilot proof.....	5
4.4 Controlled automatic test note.....	6
4.5 Decision.....	6
5. Defect ownership and acceptance matrix.....	6
6. Group 1 entry baseline.....	7
7. Developer validation after Group 0.....	7

1. Executive conclusion

Group 0 is complete. The two blocking ownership questions from the completed test report have been resolved:

1. The earlier 179-entry scan and the current 173-entry scan do not reveal six lost Website Analytics strings. The six retired keys belong exclusively to the Studio's former source-modifying menu integration. Those menu controls are absent from both the pristine GA4 source and the current component pilot. The current 173-entry result is therefore the correct like-for-like form-resource baseline.
2. The TC-18 repeating offline dialog is owned by Website Analytics, not by `TDATLanguageManager`. The pristine application and component pilot contain identical startup-authentication, auto-update, and update-button procedures. The unhandled network call occurs in the pristine application's startup authentication timer before localization participates.

The localization project should not modify Website Analytics to repair TC-18. Website Analytics needs its own controlled network-failure correction. Translation Studio work can now proceed to Group 1: runtime lifecycle and state ownership.

2. Preservation and safety record

Before Group 0 work, complete pre-change backups were created:

- `<BackupDrive>:\Delphi App Translation Backup\Pre-Group-0-Translation-Studio-2026-08-09_191054`
- `<BackupDrive>:\FMXPilot - Component Test Backup\Pre-Group-0-Component-Test-2026-08-09_191054`

The following evidence remained available in place:

- Current development catalog: `C:\Projects\FMXPilot - Component Test\Localization\Development\FMXPilot.es-ES.translation-project.json`
- Current Spanish runtime pack: `C:\Projects\FMXPilot - Component Test\Localization\Languages\es-ES.json`
- Generated component kit: `C:\Projects\Delphi App Translation\export\component-integration\FMXPilot`
- Earlier test archive: `C:\TestArchives\TranslationPilot`
- Pristine GA4 source: `C:\Projects\FMXPilot - Pristine`

The disposable pilot was clean in Git before investigation. The Studio contained known pre-existing untracked generated/export files; those files were not staged or altered as part of Group 0.

3. Reconciliation of 179 and 173 entries

3.1 Compared catalogs

Earlier 179-entry catalog

C:\TestArchives\TranslationPilot\Localization\Development\FMXPilot.es-ES.translation-project.json

SHA-256: 246f386c1c60dc2f4d18d9d975a660bb005587f0ed62b0713eef60d647650709

Current 173-entry catalog

C:\Projects\FMXPilot - Component Test\Localization\Development\FMXPilot.es-ES.translation-project.json

SHA-256: cb0fc82490dc644eeca60d9577c13c887bb41cd013bf9e81db056e6b1c90a4fb

3.2 Stable-key comparison

- Earlier entries: 179
- Current entries: 173
- Keys present only in the earlier catalog: 6
- Keys present only in the current catalog: 0
- Common keys with changed source text: 0
- Common keys with changed runtime classification: 0

The six earlier-only keys are:

1. frmMainDashboard.datLanguage_en_US.Text = English
2. frmMainDashboard.datLanguage_es_ES.Text = Español
3. frmMainDashboard.datLanguage_fr_FR.Text = Français
4. frmMainDashboard.datTranslationExitMenuItem.Text = E&xit
5. frmMainDashboard.datTranslationFileMenu.Text = &File
6. frmMainDashboard.mnuLanguage.Text = &Language

3.3 Source proof

All six controls exist in the archived source-mutating pilot under:

- C:\TestArchives\TranslationPilot\FMXPilot - Old-Component Test\FMXPilot.MainForm.fmx
- C:\TestArchives\TranslationPilot\FMXPilot - Old-Component Test\FMXPilot.MainForm.pas

None of the six controls exists in:

- the pristine GA4 project; or
- the current component pilot.

They were generated by the former integration design that inserted a File/Exit/Language menu into the target application. The approved component architecture intentionally stopped modifying target forms

and source. Its developer-owned selector is a component control and its language choices come from runtime packs, so those obsolete menu captions should not reappear in the scan.

3.4 Decision

Close DAT-004 as **resolved - explained architecture change, not scanner regression**. Keep 173 as the frozen baseline for the current source state. The untranslated dynamic/chart/status text observed during testing remains valid work under DAT-002, DAT-003, and DAT-005; it is unrelated to the six removed menu keys.

4. TC-18 ownership analysis

4.1 Observed failure

During TC-18, Website Analytics was run without internet connectivity. It displayed `Error sending data: (12007) The server name or address could not be resolved`. Dismissing the modal dialog led to another occurrence, making the application effectively unusable.

4.2 Static execution trace

Website Analytics starts `AuthStartupTimer`. When a saved Google refresh token is available, `TfrmMainDashboard.AuthStartupTimerTimer` calls:

```
dmAuthentication.RefreshAccessToken(OAuthClientId, OAuthClientSecret)
```

`RefreshAccessToken` creates a `THttpClient` and posts to:

```
https://oauth2.googleapis.com/token
```

That HTTP call is not protected by a local `try..except`. When DNS or networking fails, the exception escapes the timer event and reaches FMX's application exception handler, which displays the modal dialog. The timer is not disabled on that exception path, so it can call the same operation again.

The later dashboard update path does have an exception handler, but that does not protect the earlier startup-token refresh.

4.3 Pristine-versus-pilot proof

The following complete procedures are textually identical in the pristine project and the component pilot:

- `TfrmMainDashboard.AuthStartupTimerTimer`
- `TfrmMainDashboard.AutoUpdateTimerTimer`
- `TfrmMainDashboard.btnUpdateClick`

The pristine source already contains the unguarded saved-token refresh. The component integration added the language manager, language combo box, and label, but did not create or alter this network/timer behavior. No localization runtime code appears in the failing call chain.

4.4 Controlled automatic test note

An automatic isolation attempt was made with a guarded cleanup block. Windows denied permission to disable the Wi-Fi adapter, so the executable was not launched in a newly altered network state. The cleanup block still ran and verified `Wi-Fi` was `Up`. No network setting was left changed.

The supplied TC-18 screenshot establishes the runtime symptom. The identical pristine and pilot procedures, plus the direct call-chain trace, establish ownership without changing or recompiling either project. A further live isolation run is not required to decide which codebase owns the defect.

4.5 Decision

Assign DAT-001 to **Website Analytics**. Keep it visible as a pilot release blocker, but do not repair it inside Translation Studio or `TDATLanguageManager`.

Website Analytics acceptance behavior should be:

- catch network exceptions inside startup authentication;
- disable or back off the responsible timer after failure;
- show no repeating modal-dialog loop;
- retain the last valid dashboard state;
- provide a clear non-modal status and manual retry; and
- resume normally when connectivity returns.

5. Defect ownership and acceptance matrix

Defect	Proven owner	Group	Acceptance test
DAT-001 repeating offline dialogs	Website Analytics networking/authentication	Separate GA4 correction	Offline startup produces no repeated modal dialogs; retry succeeds after reconnection.
DAT-002 localization overwrites live state	Language-manager classification plus host runtime-message integration	1	Language switching preserves actual connection state, selected dates, and live values.
DAT-003 mixed-language restart	Manager lifecycle plus host post-load/generated assignments	1	Restored Spanish is consistent after startup and after the first data refresh.
DAT-004 179 versus 173	Resolved; former menu integration	0 complete	Stable-key diff remains exactly the six retired menu controls; current baseline stays 173.
DAT-005 untranslated generated/static text	Scanner/runtime templates plus host owner-drawn integration	2	Every known human-language string is translated, excluded with a reason, or registered as a template.

Defect	Proven owner	Group	Acceptance test
DAT-006 crowding/truncation	Target FMX layout; Studio expansion warnings	4	Essential Spanish text wraps or expands without clipping at supported sizes.
DAT-007 English HTML help	Separate help localization workflow	Deferred / 4	Help selects a locale-specific package, or clearly falls back to English.
DAT-008 mojibake language names	Studio encoding and catalog I/O	4	UTF-8 round trips preserve native names and representative scripts.
DAT-009 Studio UI defects	Translation Studio FMX forms	4	Page-by-page minimum/maximized review shows no overlap, truncation, or stale status.
DAT-010 guide ambiguities	Translation Studio documentation	5	A clean-room tester completes all cases without oral correction.

6. Group 1 entry baseline

Group 1 should begin with no target-source mutation and preserve the component architecture. Its first controlled work package should:

1. define static, dynamic, template, data, identifier, and excluded classifications;
2. prevent pack application from replacing live state labels;
3. define keyed runtime messages with placeholders;
4. establish a deterministic FMX startup and language-reapply lifecycle; and
5. add regression tests for immediate switching, saved-language restart, connection state, and date-range state.

No scanner expansion should begin until these runtime ownership rules are executable and tested. Otherwise, broader scanning could add more strings that the runtime should never overwrite.

7. Developer validation after Group 0

No application rebuild or manual test is required for this investigation group.

The developer should:

1. Read this report, especially Sections 3 through 6.
2. Confirm that the six retired menu keys should remain retired under the component architecture.
3. Confirm that the Website Analytics offline-loop repair remains separate from Translation Studio Group 1.
4. Approve Group 1 when ready.

The next hands-on application testing begins only after Group 1 implementation. Exact steps will be supplied at that time.

Appendix A. License Information

Unless a particular file or third-party component states otherwise, Delphi App Translation Studio and its project documentation are licensed under the Apache License, Version 2.0. The license permits broad use while preserving attribution, notices, and the stated warranty limitations.

License grant

Subject to the complete license terms, you may use, reproduce, modify, prepare derivative works from, publicly display, publicly perform, sublicense, and distribute the licensed material. The Apache License 2.0 also includes an express patent license from contributors for their applicable contributions.

Important conditions

- Give recipients a copy of the Apache License 2.0 when distributing covered material.
- State significant changes made to modified files.
- Retain applicable copyright, patent, trademark, attribution, and NOTICE information.
- Do not use project or contributor names and trademarks as an endorsement except as the license permits.

Warranty and liability

The licensed material is provided on an AS IS basis, without warranties or conditions of any kind. The complete Apache License 2.0 controls the warranty, liability, contribution, redistribution, and patent provisions.

Your applications and generated output

Using the Studio does not transfer ownership of the Delphi application being translated. The application developer remains responsible for the license and distribution terms of the target application, translated content, generated language packs, and any third-party components. Studio runtime or component source included with a project remains subject to the Apache License 2.0 unless its file states different terms. Third-party software retains its own license.

Full legal text

The complete controlling license is the LICENSE file in the repository and source distribution root. An official reference copy is available at <https://www.apache.org/licenses/LICENSE-2.0>. If this summary and the complete license differ, the complete Apache License 2.0 text controls.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this work except in compliance with the License. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.