

QA Compiler Suite - User Guide

QA Compiler Suite - User Guide

VERSION	2.0	PLATFORM	WINDOWS	STATUS	PRODUCTION
---------	-----	----------	---------	--------	------------

Your Complete Guide to QA Localization Workflow

Table of Contents

Section	Description
Installation	Build and setup
Workflows Overview	Weekly vs Daily tasks
Weekly Workflow	Generate → Transfer → Build
Daily Workflow	Download and organize tester files
Main Interface	Understanding the GUI
1. Generate Datasheets	Create LQA worksheets
2. Transfer QA Files	Merge tester work
3. Build Master Files	Compile master documents
4. Coverage Analysis	Check translation coverage
5. System Localizer	Localize System sheets
Folder Structure	Where files go
Folder Naming Convention	How to name tester folders
Troubleshooting	Common issues

□ Installation

Build Your Own Executable

Each user builds their own executable to match their Perforce drive location.

Step Action

- 1 **Extract** the QACompilerNEW.zip to a folder
- 2 **Run** build_exe.bat
- 3 When prompted, **enter your drive letter** (F, D, E, etc.)
- 4 Wait for build to complete
- 5 Find executable in dist\QACompiler\QACompiler.exe

Enter drive letter (F/D/E/etc.) [F]: D

Why Build Yourself?

- **Different drives:** Perforce can be on F:, D:, E: etc.
- **Correct paths:** Build process configures paths for YOUR system
- **No manual editing:** Drive selection is automatic

Requirements

Requirement Details

Python	3.8 or higher
pip	Comes with Python
Perforce	Must be synced to your machine

After Building

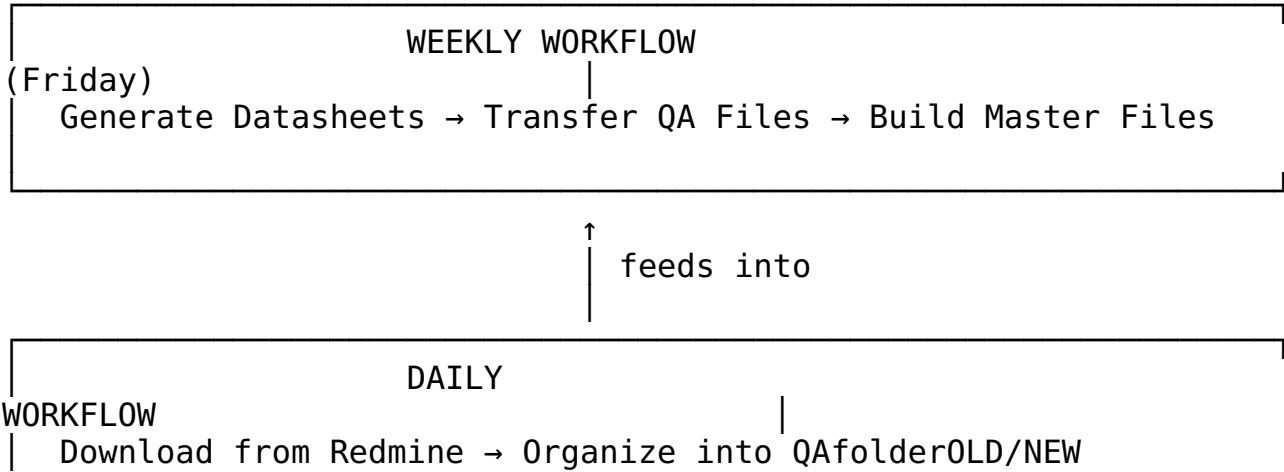
Copy the entire dist\QACompiler\ folder to your preferred location:

```
C:\Tools\QACompiler\  
├── QACompiler.exe      ← Double-click to run  
├── QAfolder\  
├── QAfolderOLD\  
├── QAfolderNEW\  
└── ...
```

Workflows Overview

The QA Compiler supports two main workflows:

Workflow	Frequency	Purpose
Weekly (Friday Refresh)	Every Friday	Refresh all datasheets with new game data
Daily	Every day	Process tester submissions



□ Weekly Workflow (Friday Refresh)

Every Friday, refresh the QA files with the latest game data.

Step 1: Generate Datasheets

Creates fresh LQA worksheets from game XML data.

Action Details

Click **[Generate Selected]** (or select specific categories)

Output GeneratedDatasheets/ folder

When Game data has been updated

Step 2: Transfer QA Files

Merges tester work from OLD/NEW folders into QAfolder.

Action Details

Ensure QAfolderOLD/ and QAfolderNEW/ have tester folders

Click **[Transfer QA Files]**

Output QAfolder/ (combined)

Step 3: Build Master Files

Compiles everything into final master documents.

Action Details

Click **[Build Master Files]**

Output Masterfolder_EN/ and Masterfolder_CN/

Includes Progress tracker (_TRACKER.xlsx)

Weekly Workflow Summary

- | | |
|------------------------|---------------------------------------|
| 1. Generate Datasheets | → Fresh worksheets from game XML |
| 2. Transfer QA Files | → Merge tester work into QAfolder |
| 3. Build Master Files | → Compile into Master files + Tracker |

□ Daily Workflow

Every day, collect and organize tester submissions.

Step 1: Download from Redmine

Testers upload their QA files to Redmine. Download them daily.

Source What to Download

Redmine Tester-submitted QA folders

Format 이름_Category folders (see naming convention below)

Step 2: Organize into Folders

Place downloaded folders into the appropriate location:

Folder What Goes Here

QAfolderOLD/ **Previous round** - tester's last submitted work

QAfolderNEW/ **Current round** - tester's new empty datasheets

Step 3: Run Transfer (When Ready)

Once you have both OLD and NEW files for a category:

1. Click **[Transfer QA Files]**
2. Combined output appears in QAfolder/

□ QAfolder Behavior (Important!)

The QAfolder/ is the **master collection** of all QA work.

Golden Rules

Rule	Explanation
Never delete manually	Files are managed by Transfer process
Only add/edit	New categories get added, existing ones get updated
Auto-updated	Transfer process handles all merging
Keeps history	Completed categories stay until next refresh

How It Works

Before Transfer:

QAfolder/

```
|—— 김민영_Quest/      ← Completed last week, KEEP IT
|—— 박지훈_Knowledge/ ← Completed last week, KEEP IT
|—— (empty for new categories)
```

After Transfer:

QAfolder/

김민영_Quest/	← Still there (untouched)
박지훈_Knowledge/	← Still there (untouched)
이수진_Item/	← NEW - just transferred
최영희_Region/	← NEW - just transferred

Why This Matters

- **Completed work is preserved** until weekly refresh
- **No accidental deletions** - Transfer only adds/updates
- **Incremental updates** - Add new categories as testers finish

Main Interface

QA Compiler Suite v2.0		
<div> <div>1. Generate Datasheets</div> <div> <div> <input checked="" type="checkbox"/> Quest <input checked="" type="checkbox"/> Knowledge <input checked="" type="checkbox"/> Item </div> <div> <input checked="" type="checkbox"/> Region <input checked="" type="checkbox"/> System <input checked="" type="checkbox"/> Character </div> <div> <input checked="" type="checkbox"/> Skill <input checked="" type="checkbox"/> Help <input checked="" type="checkbox"/> Gimmick </div> </div> <div> <div>[Select All]</div> <div>[Deselect All]</div> <div>[Generate Selected]</div> </div> </div>		
<div> <div>2. Transfer QA Files</div> <div>[Transfer QA Files]</div> </div>		
<div> <div>3. Build Master Files</div> <div>[Build Master Files]</div> </div>		
<div> <div>4. Coverage Analysis</div> <div>[Run Coverage Analysis]</div> </div>		
<div> <div>5. System Sheet Localizer</div> <div>[Localize System Sheet]</div> </div>		
<div> <div>Status: Ready</div> <div>[=====]</div> </div>		

1. Generate Datasheets

Purpose: Create fresh LQA worksheets from game XML data.

Category Guide

Category	Contains	Output
Quest	Main story, faction, daily quests	QuestData_Map_All/
Knowledge	Encyclopedia entries	Knowledge_LQA_All/

Category	Contains	Output
Item	Items, equipment, consumables	ItemData_Map_All/
Region	Areas, locations, POIs	Region_LQA_v3/
System	UI text, menus	(via Skill+Help merge)
Character	NPCs, monsters	Character_LQA_All/
Skill	Player abilities	Skill_LQA_All/
Help	Tutorial, tips	GameAdvice_LQA_All/
Gimmick	Interactive objects	Gimmick_LQA_Output/

Output Excel Columns

Column	Description	Editable?
Original (KR)	Korean source text	<input type="checkbox"/> No
English (ENG)	English translation	<input type="checkbox"/> No
Translation	Target language text	<input type="checkbox"/> No
STATUS	Issue status dropdown	<input type="checkbox"/> Yes
COMMENT	Tester notes	<input type="checkbox"/> Yes
STRINGID	Unique identifier	<input type="checkbox"/> No
SCREENSHOT	Screenshot reference	<input type="checkbox"/> Yes

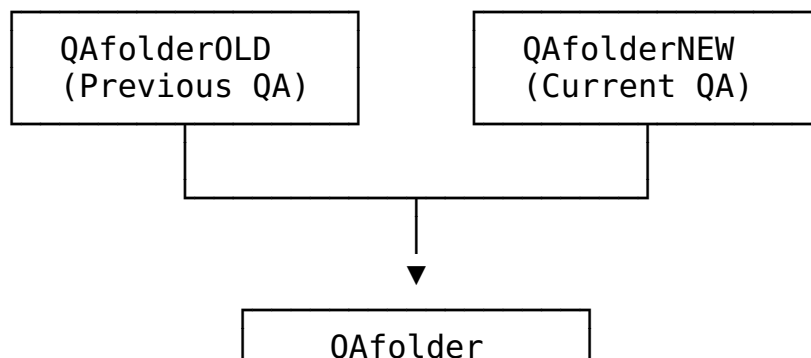
Tester STATUS Options

Status	Meaning	Color
ISSUE	Problem found - needs fix	<input type="checkbox"/> Red
NO ISSUE	Checked, looks good	<input type="checkbox"/> Green
BLOCKED	Cannot test	<input type="checkbox"/> Yellow
KOREAN	Still in Korean	<input type="checkbox"/> Orange

2. Transfer QA Files

Purpose: Merge tester work from OLD/NEW folders into QAfolder.

The Flow



(Combined)

How to Use

Step Action

- 1 Place OLD tester folders in QAfolderOLD/
 - 2 Place NEW tester folders in QAfolderNEW/
 - 3 Click **[Transfer QA Files]**
 - 4 Combined output appears in QAfolder/
-

3. Build Master Files

Purpose: Compile all QA files into master documents with progress tracking.

Category Merging

Some categories are **merged** into combined master files:

Input Category	Output Master File
Quest	Master_Quest.xlsx
Knowledge	Master_Knowledge.xlsx
Item	Master_Item.xlsx
Region	Master_Region.xlsx
Character	Master_Character.xlsx
Skill	Master_System.xlsx ← <i>merged</i>
Help	Master_System.xlsx ← <i>merged</i>
Gimmick	Master_Item.xlsx ← <i>merged</i>

Output Structure

```
Masterfolder_EN/
├── Master_Quest.xlsx
├── Master_Knowledge.xlsx
├── Master_Item.xlsx           ← includes Gimmick
├── Master_Region.xlsx
├── Master_System.xlsx        ← includes Skill + Help
├── Master_Character.xlsx
├── _TRACKER.xlsx            ← Progress tracking
└── Images/
```

Progress Tracker

The _TRACKER.xlsx contains:

Sheet	Shows
DAILY	Day-by-day progress per tester
TOTAL	Overall statistics and rankings
_DAILY_DATA	Raw data (hidden)

Automatic Row Hiding

Rows are automatically hidden based on two status columns:

TESTER STATUS (TESTER_STATUS_{User} - hidden column)

This is the **tester's original status** (from their QA work):

Status	Hidden?	Reason
ISSUE	<input type="checkbox"/> No	Active issue - needs attention
BLOCKED	<input type="checkbox"/> Yes	Tester couldn't test
KOREAN	<input type="checkbox"/> Yes	Still in Korean
NO ISSUE	<input type="checkbox"/> Yes	No problem found

MANAGER STATUS (STATUS_{User} - visible column)

This is the **manager's review status** (dropdown in Master file):

Status	Hidden?	Reason
FIXED	<input type="checkbox"/> Yes	Issue resolved
NON-ISSUE	<input type="checkbox"/> Yes	Not actually an issue
REPORTED	<input type="checkbox"/> No	Reported to dev team
CHECKING	<input type="checkbox"/> No	Under investigation
(empty)	<input type="checkbox"/> No	Pending manager review

Summary: Only ISSUE rows that haven't been resolved by manager are visible.

4. Coverage Analysis

Purpose: Calculate translation coverage.

Creates Coverage_Report_YYYYMMDD_HHMMSS.xlsx with:

Sheet	Contents
Coverage Report	Strings covered per category
Word Count	Korean + Translation word counts

□ 5. System Localizer

Purpose: Create localized System sheets for all languages.

Output

```
System_LQA_All/  
├── System_ENG.xlsx  
├── System_DEU.xlsx  
├── System_FRA.xlsx  
├── System_JPN.xlsx  
└── ... (all languages)
```

Matching Process

Step 1: StringID → Korean → Target Language (most accurate)

Step 2: English Text → Korean → Target Language (fallback)

□ Folder Structure

```
QACompiler/  
├── QACompiler.exe           ← Main application  
├── QAfolderOLD/             ← Previous round tester files  
│   ├── 김민영_Quest/  
│   └── 박지훈_Item/  
├── QAfolderNEW/             ← Current round tester files  
│   ├── 김민영_Quest/  
│   └── 박지훈_Item/  
├── QAfolder/                 ← Combined (auto-generated)  
│   ├── 김민영_Quest/  
│   └── 박지훈_Item/  
├── GeneratedDatasheets/     ← From "Generate Datasheets"  
├── Masterfolder_EN/         ← English master output  
│   ├── Master_Quest.xlsx  
│   ├── TRACKER.xlsx  
│   └── Images/  
└── Masterfolder_CN/         ← Chinese master output
```

Folder Naming Convention

Tester folders must follow this format: 이름_Category

Format Example Explanation

이름_Category 김민영_Quest Name + underscore + Category

Valid Examples

Folder Name	Tester	Category
김민영_Quest	김민영	Quest
박지훈_Item	박지훈	Item
이수진_Knowledge	이수진	Knowledge
최영희_Region	최영희	Region
John_Quest	John	Quest

Valid Categories

Category

Quest
Knowledge
Item
Region
System
Character
Skill
Help
Gimmick

Rules

Rule	Correct	Wrong
Single underscore	김민영_Quest	김_민_영_Quest
Category at end	김민영_Quest	Quest_김민영
Exact category name	김민영_Quest	김민영_quest

❏ Troubleshooting

❏ “Generator modules not yet implemented”

Cause: Generator files missing or import failed.

Solution: 1. Verify all files present 2. Check generators/ folder exists 3. Re-extract from package

❏ “No datasheets found in GeneratedDatasheets”

Cause: Coverage analysis needs datasheets first.

Solution: 1. Run “Generate Datasheets” first 2. Verify files in GeneratedDatasheets/ 3. Then run Coverage Analysis

❑ Path errors mentioning wrong drive

Cause: Executable built for different drive.

Solution: 1. Re-run build_exe.bat 2. Enter YOUR drive letter when prompted 3. Use the new executable

❑ Excel file corrupted

Cause: Process interrupted during write.

Solution: 1. Delete corrupted file 2. Run operation again 3. Don’t close app while progress bar active

❑ Folder not recognized

Cause: Folder name doesn’t match 이름_Category format.

Solution: 1. Check folder name format: 김민영_Quest 2. Verify category is valid (Quest, Item, etc.) 3. Use single underscore only

❑ Support

Need	Contact
Bug reports	Your QA Lead
Feature requests	Development Team
Access issues	IT Department

QA Compiler Suite v2.0

Making localization QA easier, one datasheet at a time.

