Desktop Icon Backup Manager

User Manual

Software Version: 0.7.0

Developed by: **mapi68**

January 6, 2026

Contents

1 Introduction **4**

1.1 Key Features 4

1.2 System Requirements 4

2 Installation and Startup **4**

2.1 Installation 4

2.2 First Run 5

3 Main Interface **5**

3.1 Interface Overview 5

3.2 Main Action Buttons 5

3.2.1 💾 SAVE QUICK BACKUP 5

3.2.2 ↺ RESTORE LATEST 6

3.2.3 ☰ BACKUP MANAGER 7

4 File Menu **7**

4.1 Scramble Desktop Icons (Random) 7

4.2 Remove All Backups 8

4.3 Exit 8

5 Settings Menu **8**

5.1 Start Minimized to Tray 8

5.2 Auto-Save on Exit 9

5.3 Auto-Restore on Startup 9

5.4 Enable Adaptive Scaling on Restore 10

5.4.1 How Adaptive Scaling Works 10

5.4.2 When Scaling Is Applied 11

5.4.3 Log Messages for Scaling 11

5.5 Minimize to Tray on Close 11

5.6 Automatic Backup Cleanup Limit 12

6 Advanced Backup Management **12**

6.1 Backup Manager Window 12

6.1.1 Search and Filter Bar 12

6.1.2 Backup List Columns 13

6.1.3 Visual Preview Panel 13

6.1.4 Interactive Tooltip Preview 14

6.1.5 Information Panel 14

6.1.6 Available Operations 14

6.1.7 Comparison with Latest Backup 14

6.2 Backup File Format 15

6.2.1 File Naming Convention 15

6.2.2 JSON File Structure (Complete) 15

7 System Tray Integration **16**

7.1 System Tray Context Menu 16

7.2 Tray Notifications 17

8 Keyboard Shortcuts **17**

9 Help Menu **17**

9.1 Online User Manual . . . 17
9.2 About Dialog . . . 18

10 Troubleshooting 18
10.1 Common Issues . . . 18
10.1.1 Error: "Unable to find desktop ListView control" . . . 18
10.1.2 Icons Restored to Wrong Positions . . . 18
10.1.3 Backup File Not Found . . . 19
10.1.4 Program Crashes on Startup . . . 19
10.1.5 Scramble Operation Fails . . . 20
10.1.6 Settings Not Persisting . . . 20
10.2 Diagnostics . . . 20
10.3 Performance Issues . . . 20
10.3.1 Slow Save/Restore Operations . . . 20
10.3.2 High Memory Usage . . . 21

11 Best Practices 21
11.1 Optimal Backup Strategy . . . 21
11.2 Backup Naming Best Practices . . . 21
11.3 Multi-Monitor Management . . . 22
11.4 Resolution Change Scenarios . . . 22
11.5 Regular Maintenance . . . 22
11.6 Backup Archiving Strategy . . . 23

12 Frequently Asked Questions (FAQ) 23
12.1 General Questions . . . 23
12.2 Feature Questions . . . 24
12.3 Technical Questions . . . 24

13 Technical Information 25
13.1 Software Architecture . . . 25
13.2 Remote Memory Access Process . . . 25
13.3 Data Structures . . . 26
13.4 Threading Architecture . . . 26
13.5 Desktop Refresh Mechanism . . . 27
13.6 Settings Storage . . . 27
13.7 Security and Privacy . . . 28

14 Advanced Usage 28
14.1 Command Line Arguments . . . 28
14.2 Technical Implementation Details . . . 28
14.3 Example Batch Script . . . 29
14.4 Automation with Task Scheduler . . . 29
14.5 Batch Operations with JSON . . . 29

15 Development and Contribution 30
15.1 Building from Source . . . 30
15.2 Translation/Localization . . . 30

16 Technical Information 30
16.1 Software Architecture . . . 30
16.2 Remote Memory Access . . . 31
16.3 Security and Privacy . . . 31

17 License and Credits 31
17.1 Developer 31
17.2 Libraries Used 31
18 Support and Contacts 31
19 Screenshots 32

1 Introduction

Desktop Icon Backup Manager is a professional tool designed to save and restore Windows desktop icon positions with advanced features for managing multiple layouts, adaptive scaling across different resolutions, and complete automation through configurable settings.

1.1 Key Features

- **Quick Backup**: Save icon positions with optional descriptive tags
- **Backup Management**: Dedicated interface with search filtering to select, restore, or delete specific backups
- **Visual Preview**: Mini-map display of icon layouts before restoring
- **Adaptive Scaling**: Automatic adjustment of icon positions across different resolutions
- **Automatic Cleanup**: Configurable limit on the number of backups to retain (5, 10, 25, 50, or unlimited)
- **Random Scramble**: Randomize icon positions with automatic preventive backup
- **System Tray Integration**: Quick access to main functions from the system tray
- **Automation**: Auto-save on exit and auto-restore on startup options
- **Multi-language Support**: Internationalization ready with translation system

1.2 System Requirements

Minimum Requirements

- Operating System: Windows 7 or higher (fully compatible with Windows 11)
- Python 3.8+ (if running from source)
- Libraries: PyQt6, pywin32
- Disk Space: 50 MB for application + space for backups (typically 2-10 KB per backup)
- Permissions: Standard user permissions (no administrator rights required)

2 Installation and Startup

2.1 Installation

Method 1: Standalone Executable (Recommended)

1. Download the `desktop-icon-backup-manager.exe` file
2. Place the executable in a dedicated folder
3. Run the program with a double-click
4. The program will automatically create necessary folders and configuration files

Method 2: From Python Source

1. Install Python 3.8 or higher
2. Install dependencies:

```
pip install PyQt6 pywin32
```

3. Run the script:

```
python desktop-icon-backup-manager.py
```

2.2 First Run

On first run, the program automatically:

- Creates the `icon_backups` folder for backup files
- Generates the `settings.ini` configuration file in the application directory
- Verifies the presence and accessibility of desktop icons
- Places itself in the system tray (if enabled)
- Loads default settings and preferences

Important - Desktop Icons Must Be Visible

Make sure desktop icons are visible (Right-click desktop → View → Show desktop icons). If icons are hidden, the program cannot access their positions and will show an error: ”Unable to find desktop ListView control.”

3 Main Interface

3.1 Interface Overview

The main interface is divided into five main areas:

1. **Menu Bar**: Contains File, Settings, and Help menus with all advanced options
2. **Progress Bar**: Shows real-time progress during backup, restore, and scramble operations
3. **Tag Input Field**: Optional field for entering descriptive tags for quick saves
4. **Action Buttons**: Three large, color-coded buttons for primary operations
5. **Activity Log**: Displays operation status, warnings, and errors in real-time with timestamps
6. **Status Bar**: Shows current primary monitor resolution and quick operation messages

3.2 Main Action Buttons

3.2.1 💾 SAVE QUICK BACKUP

Color: Green — Shortcut: Ctrl+S

Immediately saves current desktop icon positions to a new timestamped backup file. You can add an optional descriptive tag in the field above the button for easier identification later.

Operation Sequence:

1. Disables UI buttons to prevent concurrent operations
2. Displays progress bar
3. Scans all icons present on the desktop via Windows ListView control
4. Records exact X and Y pixel coordinates for each icon

5. Saves metadata including resolution, icon count, timestamp, and description
6. Applies automatic cleanup if limit is configured (deletes oldest backups)
7. Forces desktop refresh to ensure visual consistency
8. Re-enables UI and displays completion message

Example Log Output:

```
[14:30:15] Starting new timestamped backup...
[14:30:15]   (Tag: Work Setup Final)
[14:30:15] Monitor Resolution: 1920x1080
[14:30:15] Found 12 icons. Starting scan...
[14:30:16]   Saved 12 icons to backup file ’1920x1080_20241211_143015.json’
[14:30:16]   (Description: Work Setup Final)
```

3.2.2 ↺ RESTORE LATEST

Color: Red

Restores icon positions from the most recent backup file. Before restoring, a detailed confirmation dialog displays complete backup information.

Confirmation Dialog Information:

- Full backup file name
- Saved resolution with comparison to current resolution
- Total number of icons in the backup
- Tag/description (if available)
- Backup date and time in human-readable format (YYYY/MM/DD HH:MM:SS)

Restore Process:

1. Validates backup file existence and format
2. Disables window redrawing for performance
3. Reads saved icon positions from JSON file
4. Applies adaptive scaling if enabled and resolutions differ
5. Maps saved icon names to current desktop icons
6. Updates each icon position using Windows API
7. Re-enables redrawing and forces desktop refresh
8. Reports statistics (icons restored, icons skipped)

Smart Restore Behavior

If an icon exists in the backup but not on the current desktop, it will be skipped and reported. This prevents errors when restoring backups from different system states.

3.2.3 ☰ BACKUP MANAGER

Color: Blue

Opens the comprehensive backup management window that provides:

- **Search/Filter Bar**: Real-time filtering by tag, resolution, or date
- **Backup List**: Tabular view with aligned columns for easy scanning
- **Visual Preview**: Mini-map display of icon layouts before restoring, with interactive tooltips for icon identification
- **Detailed Information**: Extended metadata display for selected backup
- **Context Menu**: Right-click for quick restore or delete actions
- **Batch Operations**: Select and delete multiple backups

4 File Menu

4.1 Scramble Desktop Icons (Random)

This function completely randomizes the positions of all desktop icons across the available screen area, creating a chaotic but fun layout.

Mandatory Pre-Scramble Backup

Before scrambling, an automatic backup is **always** created with the tag "Auto-Backup before Scramble (Random)". This ensures you can always restore the previous layout. The scramble operation will abort if this backup fails.

Detailed Procedure:

1. Select `File > Scramble Desktop Icons (Random)`
2. Confirm the operation in the warning dialog
3. Program creates mandatory backup (first 50% of progress)
4. Desktop redrawing is temporarily disabled for performance
5. Each icon is assigned random X and Y coordinates within screen bounds
6. Positions use margin buffer (100 pixels) to prevent edge clipping
7. Desktop redrawing is re-enabled (remaining 50% of progress)
8. System refresh signals are broadcast

Technical Details:

- Uses `GetSystemMetrics` to determine virtual screen dimensions
- Random positions: `random.randint(margin, screen_dimension - margin)`
- Margin prevents icons from being placed too close to screen edges
- Operation completes even if some icons fail to move

4.2 Remove All Backups

Permanently deletes all saved backup files from the `icon_backups` folder.

Warning - Irreversible Operation

This action cannot be undone. All backups will be permanently deleted from the system. You will be prompted with the exact number of files to be deleted before confirmation.

Deletion Process:

1. Counts all JSON backup files
2. Shows confirmation dialog with file count
3. Iterates through each backup file
4. Attempts deletion and logs success/failure for each
5. Reports final statistics (deleted count, failed count)
6. Shows summary dialog

4.3 Exit

Closes the application gracefully with multiple configurable behaviors:

Exit Behaviors:

- Saves current window geometry (position and size) to settings
- Creates automatic backup if ”Auto-Save on Exit” is enabled
- Respects ”Minimize to Tray on Close” setting if window is closed via X button
- Saves all pending settings changes
- Properly closes system tray icon
- Terminates console window if running as PyInstaller executable

Exit Methods:

- Menu: `File > Exit`
- Keyboard: `Ctrl+Q`
- System Tray: Right-click icon → Exit
- Window Close (X button): Behavior depends on ”Minimize to Tray” setting

5 Settings Menu

5.1 Start Minimized to Tray

When enabled, the application starts hidden in the system tray instead of showing the main window.

Use Cases:

- Users who want the program always available in background
- Systems with auto-restore enabled (no need to see the window)

- Reducing desktop clutter while maintaining quick access

Behavior:

- System tray notification: ”Application started minimized to system tray”
- Notification duration: 2 seconds
- Double-click tray icon or use context menu to show window

5.2 Auto-Save on Exit

Automatically creates a backup of current icon positions when closing the program normally (not via Task Manager or system crash).

> **Strongly Recommended**
>
> This option is highly recommended to avoid data loss from unsaved layout changes. It provides a safety net for forgetful users and ensures your latest layout is always preserved.

Technical Implementation:

- Backup is created in `closeEvent` before application exit
- Tag: ”Auto-Save on Exit”
- Respects configured automatic cleanup limit
- Does not require user confirmation
- Logs operation silently (visible only if window is open)
- If backup fails, application still exits

Interaction with Other Settings:

- Works independently of ”Minimize to Tray on Close”
- Complements ”Auto-Restore on Startup” for seamless layout preservation
- Cleanup limit prevents unlimited backup accumulation

5.3 Auto-Restore on Startup

Automatically restores the most recent backup when the application starts.

Detailed Behavior:

- Restore occurs 1 second after startup (QTimer delay)
- Uses latest backup file by timestamp
- Applies configured adaptive scaling settings
- Shows full operation log in main window
- No user confirmation required
- If restore fails, application continues normally

Important Consideration

When combined with "Auto-Save on Exit", this creates an automatic save/restore cycle. While convenient, be cautious: if icons are in wrong positions when exiting, they will be automatically restored to those wrong positions on next startup. Use Backup Manager to restore specific known-good layouts if needed.

Recommended Use Cases:

- Single-monitor setups with stable icon arrangements
- Users who frequently restart their computer
- Systems where Windows randomly moves icons (common issue)
- Combination with laptop docking/undocking workflows

5.4 Enable Adaptive Scaling on Restore

Enables intelligent scaling of icon positions when restoring a backup created at a different screen resolution.

5.4.1 How Adaptive Scaling Works

The system calculates independent scaling factors for X and Y axes based on resolution differences:

$$\text{scale}_x = \frac{\text{current width}}{\text{saved width}}$$

$$\text{scale}_y = \frac{\text{current height}}{\text{saved height}}$$

Each saved coordinate is then transformed using these factors:

$$x_{\text{new}} = \lfloor x_{\text{saved}} \times \text{scale}_x \rfloor$$
$$y_{\text{new}} = \lfloor y_{\text{saved}} \times \text{scale}_y \rfloor$$

Practical Example - Upscaling:

- Backup saved at 1920×1080 (Full HD)
- Restore on 2560×1440 (2K/QHD) monitor
- Scaling factors: $\text{scale}_x = 1.333$, $\text{scale}_y = 1.333$
- Icon at (960, 540) → restored at (1280, 720) [screen center maintained]
- Icon at (100, 100) → restored at (133, 133) [relative position preserved]

Practical Example - Downscaling:

- Backup saved at 2560×1440
- Restore on 1920×1080 monitor
- Scaling factors: $\text{scale}_x = 0.75$, $\text{scale}_y = 0.75$

- Icon at (2000, 1200) → restored at (1500, 900)
- Icon at (200, 200) → restored at (150, 150)

5.4.2 When Scaling Is Applied

Scaling Activation Conditions

Adaptive scaling only activates when ALL these conditions are met:

1. "Enable Adaptive Scaling on Restore" is checked in Settings
2. Backup file contains resolution information (newer format)
3. Current resolution differs from saved resolution
4. Both resolutions can be successfully parsed

If any condition fails, raw coordinates are used without modification.

5.4.3 Log Messages for Scaling

When Scaling Is Active:

```
[14:45:20] Adaptive Scaling enabled: Saved 1920x1080 -> Current 2560x1440
[14:45:20]   **[SCALING APPLIED]** Scaling factors: X=1.333, Y=1.333
```

When Scaling Is Skipped:

```
[14:45:20] Adaptive Scaling enabled, but resolutions match or are invalid.
           Scaling skipped.
```

When Scaling Is Disabled:

```
[14:45:20] Adaptive Scaling is disabled. Restoring raw coordinates.
```

5.5 Minimize to Tray on Close

When enabled, clicking the window's X button minimizes the program to the system tray instead of completely closing it.

Complete Closure Methods:

- Use menu: `File > Exit`
- Keyboard shortcut: `Ctrl+Q`
- Right-click tray icon: Select "Exit"

User Notification:

- Tray notification: "Application minimized to system tray. Click or double-click to restore."
- Notification duration: 2 seconds
- Window geometry is saved before minimizing

Recommended Usage

Enable this option if you want to keep the program always running in the background for quick access to tray menu functions (Quick Save, Restore Latest) without the main window occupying taskbar space.

5.6 Automatic Backup Cleanup Limit

Configures the maximum number of backup files to maintain. When this limit is exceeded after a save operation, the oldest backups are automatically deleted.

Available Options with Values:

- **Disabled (Keep All)**: No limit, all backups retained indefinitely (value: 0)
- **Keep Last 5**: Maintains only 5 most recent backups (value: 5)
- **Keep Last 10**: Maintains only 10 most recent backups (value: 10)
- **Keep Last 25**: Maintains only 25 most recent backups (value: 25)
- **Keep Last 50**: Maintains only 50 most recent backups (value: 50)

Cleanup Operation Details:

1. Cleanup runs automatically after each successful save
2. Backups are sorted by timestamp (oldest first)
3. Only files exceeding the limit are deleted
4. Deletion is logged with filename for each file
5. Current backup being created is never deleted
6. If limit is 0 (Disabled), cleanup is completely skipped

Example Log Output (Limit = 10, Current Count = 12):

```
[14:50:30] Cleanup needed: Current count (12) exceeds limit (10).
           Deleting 2 oldest file(s).
[14:50:30]   Deleted oldest backup: 1920x1080_20241201_100000.json
[14:50:30]   Deleted oldest backup: 1920x1080_20241202_120000.json
[14:50:30] Cleanup complete. Total deleted: 2 file(s).
```

Recommended Strategy by User Type

- **Casual Users**: Keep Last 5-10 (sufficient for most needs)
- **Daily Users**: Keep Last 10-25 (balances history and space)
- **Power Users**: Keep Last 25-50 (extensive history for experimentation)
- **Archivists**: Disabled (manual management, export important backups)

6 Advanced Backup Management

6.1 Backup Manager Window

The Backup Manager provides a professional interface for managing all saved backups with advanced features.

6.1.1 Search and Filter Bar

Location: Top of the window, immediately below the instruction label

Functionality:

- Real-time filtering as you type

- Case-insensitive search
- Searches across all displayed information: tag, resolution, icon count, timestamp
- Clear button (X) to quickly reset filter
- Placeholder text: "Search by tag, resolution, or date..."

Search Examples:

- Type "1920" → Shows only backups saved at 1920×* resolution
- Type "work" → Shows backups with "work" in their tag/description
- Type "2024/12" → Shows backups from December 2024
- Type "15" → Shows backups with 15 icons or saved on the 15th day

6.1.2 Backup List Columns

The list displays backups in a monospaced font (Consolas) for perfect column alignment:

1. **TAG/DESCRIPTION** (40 chars width): User-provided description, truncated with "..." if longer than 38 characters, shown in brackets [Description Here]
2. **RESOLUTION** (12 chars width): Monitor resolution at save time (e.g., "1920x1080")
3. **ICONS** (5 chars width): Number of icons in the backup, right-aligned
4. **TIMESTAMP** (variable width): Human-readable date/time in format "YYYY/MM/DD HH:MM:SS"

Example List Display:

```
TAG/DESCRIPTION              | RESOLUTION    | ICONS | TIMESTAMP
[Work Setup Final]           | 1920x1080     |    12 | 2024/12/11 14:30:15
[Gaming Layout]              | 2560x1440     |    18 | 2024/12/10 22:15:00
[Clean Desktop]              | 1920x1080     |     5 | 2024/12/09 09:00:00
[Auto-Save on Exit]          | 1920x1080     |    12 | 2024/12/08 18:45:30
```

6.1.3 Visual Preview Panel

Location: Right side of the window

Features:

- Dark background (#1a1a1a) with subtle border
- Icons displayed as blue dots (#0078d7, 3px thickness)
- Automatic scaling to fit preview area
- Updates dynamically when selecting different backups
- Shows "No Preview Available" when no backup is selected

Scaling Calculation:

- Preview uses saved resolution from backup file
- Scale X = Preview Width / Saved Width
- Scale Y = Preview Height / Saved Height
- Each icon position is multiplied by scale factors

- Positions are clamped to preview boundaries

6.1.4 Interactive Tooltip Preview

Location: Integrated within the Visual Preview Panel

New Feature: The preview panel now features an interactive tooltip system that identifies icons as you hover over them with the mouse.

- Dynamic Identification: Hovering over any blue dot (representing an icon) in the preview area will instantly display the icon's name.
- Enhanced Precision: The tooltip appears exactly at the mouse position, making it easy to identify specific icons in crowded layouts.
- Real-time Feedback: The system updates as you move the cursor across the mini-map, providing immediate visual confirmation of the backup content.

> **Pro Tip**
>
> Use this feature to verify that a backup contains specific shortcuts or files before performing a full restore, especially when dealing with multiple backups created on the same day.

6.1.5 Information Panel

Displays detailed metadata for the selected backup:

```
File: 1920x1080_20241211_143015.json
Icons: 12
Resolution: 1920x1080
Description: Work Setup Final
Timestamp: 2024-12-11T14:30:15.123456
```

6.1.6 Available Operations

Restore Operations:

- Double-click any backup in the list
- Select backup + click ”Restore Selected Layout” button
- Right-click backup → ”Restore Selected” in context menu

Delete Operations:

- Right-click backup → ”Delete Selected”
- Confirmation dialog shows before deletion
- Successful deletion refreshes the list automatically
- List view updates to reflect removal

6.1.7 Comparison with Latest Backup

The application includes an intelligent verification system that compares current desktop state with the selected backup before proceeding with the restore operation.

Information Displayed for Comparison:

- **Resolution Check**: Compares current screen resolution with the resolution stored in the backup[cite: 13, 140].
- **Icon Count**: Displays the total number of icons in the backup versus the current icons found on the desktop[cite: 140, 141].
- **Metadata Validation**: Shows the exact timestamp (YYYY/MM/DD HH:MM:SS) and optional descriptive tags of the latest file[cite: 13, 140].

Safety Verification

If the "Enable Adaptive Scaling" setting is active, the system will automatically calculate the scaling factors if it detects a resolution mismatch during this comparison phase[cite: 161, 162].

Selection Behavior:

- Single-selection mode (one backup at a time)
- Selection changes update preview and info panels immediately
- "Restore Selected Layout" button disabled when no valid selection
- Keyboard navigation supported (arrow keys, Enter to restore)

6.2 Backup File Format

Backups are stored as human-readable JSON files in the `icon_backups` subfolder.

6.2.1 File Naming Convention

Modern Format (current version):

```
[Resolution]_[Date]_[Time].json

Pattern: {width}x{height}_{YYYYMMDD}_{HHMMSS}.json
Example: 1920x1080_20241211_143015.json
```

Legacy Format (compatibility):

```
[Date]_[Time].json

Pattern: {YYYYMMDD}_{HHMMSS}.json
Example: 20241211_143015.json
```

Backward Compatibility

The program can read and restore from both modern and legacy formats. Legacy files show "N/A" for resolution in the Backup Manager.

6.2.2 JSON File Structure (Complete)

Modern Format with Full Metadata:

```
{
    "timestamp": "2024-12-11T14:30:15.123456",
    "icon_count": 12,
    "description": "Work Setup Final",
    "display_metadata": {
        "monitor_count": 2,
        "primary_resolution": "1920x1080",
        "screens": [
            {
                "id": 0,
                "name": "\\.\DISPLAY1",
                "width": 1920,
                "height": 1080,
                "pixel_density": 1.0
            },
            {
                "id": 1,
                "name": "\\.\DISPLAY2",
                "width": 1920,
                "height": 1080,
                "pixel_density": 1.0
            }
        ]
    },
    "icons": {
        "This PC": [100, 200],
        "Recycle Bin": [100, 350],
        "Documents": [100, 500],
        "Downloads": [100, 650],
        "Pictures": [100, 800],
        "Music": [100, 950],
        "Videos": [250, 200],
        "Desktop": [250, 350],
        "Network": [250, 500],
        "Control Panel": [250, 650],
        "User Files": [250, 800],
        "Shortcuts": [250, 950]
    }
}
```

Legacy Format (Old Backups):

```
{
    "This PC": [100, 200],
    "Recycle Bin": [100, 350],
    "Documents": [100, 500],
    ...
}
```

7 System Tray Integration

7.1 System Tray Context Menu

Right-click on the system tray icon to access quick functions:

- **Quick Save**: Creates immediate backup with tag "Quick Save (Tray)"

- **Restore Latest**: Restores most recent backup without confirmation
- **Show Window**: Brings main window to front and activates it
- **Exit**: Completely closes application (bypasses ”Minimize to Tray” setting)

Icon Activation Behaviors:

- **Double-click**: Shows main window (same as ”Show Window”)
- **Single-click**: No action (prevents accidental triggers)
- **Right-click**: Opens context menu

7.2 Tray Notifications

The program displays system tray notifications for important events:

When Window Is Hidden:

- Operation completion messages (Save/Restore/Scramble successful)
- Critical errors during background operations
- Warning messages requiring user attention

Always Displayed:

- Application started minimized to tray
- Application minimized to tray from window close
- Quick operation confirmations from tray menu

Notification Duration:

- Standard messages: 2000 milliseconds (2 seconds)
- Warning/Error messages: 5000 milliseconds (5 seconds)

8 Keyboard Shortcuts

Shortcut	**Action**	**Description**
`Ctrl+S`	Quick Save	Saves current layout with optional tag
`Ctrl+Q`	Exit Application	Closes program completely

Table 1: Available keyboard shortcuts

Future Enhancements

Additional keyboard shortcuts may be added in future versions based on user feedback. Consider suggesting useful shortcuts through the project repository.

9 Help Menu

9.1 Online User Manual

Opens the complete PDF user manual in your default browser.

URL: `https://mapi68.github.io/desktop-icon-backup-manager/manual.pdf`

Manual Contents:

- Complete feature documentation
- Step-by-step tutorials
- Troubleshooting guides
- Best practices and recommendations
- Technical reference information

9.2 About Dialog

Displays program information including:

- Program name and description
- Complete feature list with descriptions
- Current version number
- Developer information (mapi68)
- Quick reference to key features

10 Troubleshooting

10.1 Common Issues

10.1.1 Error: "Unable to find desktop ListView control"

Cause: Desktop icons are hidden or the ListView control is not accessible.

Solution:

1. Right-click on empty area of desktop
2. Navigate to `View > Show desktop icons`
3. Verify checkmark appears next to "Show desktop icons"
4. Restart the program
5. If error persists, check Windows Explorer is running

Advanced Troubleshooting:

- Restart Windows Explorer via Task Manager
- Check if third-party desktop replacement software is interfering
- Verify no group policies are hiding desktop icons
- Run program as administrator (if standard permissions fail)

10.1.2 Icons Restored to Wrong Positions

Cause: Resolution change, different monitor configuration, or disabled scaling.

Solutions:

1. Enable "Enable Adaptive Scaling on Restore" in Settings menu

2. Create separate backups for each monitor configuration with descriptive tags
3. Verify current resolution matches saved resolution in backup details
4. Check backup was created on same physical monitor setup
5. Use Backup Manager preview to verify layout before restoring

Multi-Monitor Specific Issues:

- Ensure same monitors are connected in same positions
- Check Windows display arrangement matches saved configuration
- Primary monitor must be same as when backup was created
- Monitor orientation (landscape/portrait) must match

10.1.3 Backup File Not Found

Cause: Corrupted or deleted backup file, missing `icon_backups` folder, file permissions.

Solution:

1. Verify `icon_backups` folder exists in program directory
2. Check folder contains .json files
3. Verify read/write permissions on folder and files
4. Open JSON file in text editor to check for corruption
5. Restore from Windows backup if available
6. Check Recycle Bin for accidentally deleted backups

JSON Validation:

- Open file in text editor (Notepad++, VS Code)
- Verify valid JSON format (matching braces, proper quotes)
- Check file is not empty (minimum valid backup is ∼100 bytes)
- Use online JSON validator if unsure about format

10.1.4 Program Crashes on Startup

Possible Causes and Solutions:

Corrupted settings.ini:

1. Delete `settings.ini` file from program directory
2. Restart program (will recreate with defaults)
3. Reconfigure your preferences

Auto-Restore failure:

1. Disable ”Auto-Restore on Startup” by editing settings.ini manually
2. Set `auto_restore_on_startup=false`
3. Restart program and investigate problematic backup

Missing dependencies (source installation):

1. Reinstall PyQt6: `pip install --upgrade PyQt6`
2. Reinstall pywin32: `pip install --upgrade pywin32`
3. Check Python version is 3.8 or higher

10.1.5 Scramble Operation Fails

Cause: Pre-scramble backup failure, insufficient permissions, or desktop access issues.

Solution:

1. Check Activity Log for specific error message
2. Verify sufficient disk space for backup creation
3. Ensure `icon_backups` folder is writable
4. Try manual backup first to test functionality
5. Close any programs that might lock desktop access

10.1.6 Settings Not Persisting

Cause: Write permissions, settings.ini location, or application not closing properly.

Solution:

1. Check settings.ini exists in application directory
2. Verify write permissions on settings.ini file
3. Use `File > Exit` or Ctrl+Q instead of Task Manager
4. Don't force-close during ”Saving settings...” operations
5. Run as administrator if in protected directory

10.2 Diagnostics

For Advanced Diagnostics:

1. Execute the problematic operation
2. Copy complete log content (Ctrl+A in log area, Ctrl+C)
3. Check for specific error messages
4. Note exact timestamp of error
5. Verify settings.ini for incorrect configurations
6. Check Windows Event Viewer for system-level errors

10.3 Performance Issues

10.3.1 Slow Save/Restore Operations

Normal Expected Times:

- Save operation: 1-3 seconds for 10-20 icons
- Restore operation: 2-5 seconds for 10-20 icons
- Scramble operation: 3-6 seconds total (includes backup)

If Operations Are Slower:

- Check CPU usage (Task Manager) - high usage indicates issue
- Verify disk is not at 100% usage (slow HDD can cause delays)
- Scan for malware/viruses affecting system performance
- Close resource-intensive applications during operations
- Consider SSD upgrade if using mechanical hard drive

10.3.2 High Memory Usage

Normal Memory Usage:

- Idle: 50-80 MB
- During operation: 80-120 MB
- With large backup list: 100-150 MB

If Memory Is Excessive:

- Close and reopen program (clears accumulated memory)
- Check for memory leaks (usage grows over time without activity)
- Report issue with version number and system specs

11 Best Practices

11.1 Optimal Backup Strategy

Recommended Configuration for Most Users

- Enable ”Auto-Save on Exit” for safety net
- Set cleanup limit to 10-25 backups (balances history and space)
- Create manual backups with descriptive tags before major changes
- Use descriptive tags like ”Before Windows Update”, ”Gaming Setup”, etc.
- Test restore operation occasionally to verify backups work

11.2 Backup Naming Best Practices

Effective Tag Examples:

- **Descriptive**: ”Work Setup - Dual Monitor”, ”Gaming - Single Screen”
- **Temporal**: ”Before Windows Update 2024-12”, ”End of Year Layout”
- **Purpose**: ”Clean Desktop Minimal”, ”Development Environment”
- **Event-based**: ”Pre-Hardware Upgrade”, ”After Monitor Replacement”

Avoid:

- Generic tags: ”backup1”, ”test”, ”new”
- Too long descriptions (truncated after 38 characters)
- Special characters that might cause issues: \/ : * ? ” ¡ ¿ —

11.3 Multi-Monitor Management

For systems with changing monitor configurations:

1. Create separate backups for each configuration with clear tags
2. Tag examples: "1 Monitor - Laptop Only", "2 Monitors - Office Setup", "3 Monitors - Home Studio"
3. Disable auto-restore if frequently switching configurations
4. Always verify current monitor setup before restoring
5. Use Backup Manager preview to visually confirm layout
6. Consider creating backup before connecting/disconnecting monitors

Docking Station Workflows:

- **Backup "Laptop Undocked"**: Before connecting to dock
- **Backup "Laptop Docked"**: After connecting monitors
- Use Backup Manager to quickly switch between configurations
- Consider enabling adaptive scaling for similar resolutions

11.4 Resolution Change Scenarios

When Upgrading Monitor:

1. Create backup with descriptive tag before upgrade
2. Note old resolution in tag: "Old Monitor 1920x1080"
3. After upgrade, test restore with scaling enabled
4. Adjust manually if needed, then save new backup
5. Keep old backup for reference or if reverting

When Using Laptop with Different External Monitors:

- Create backup for each location: "Home Office", "Work Office", "Portable"
- Include resolution in tag if significantly different
- Test adaptive scaling effectiveness for each pair
- Keep laptop-only backup as fallback

11.5 Regular Maintenance

Weekly:

- Review recent backups in Backup Manager
- Delete any test or unnecessary backups
- Verify auto-save/restore working as expected

Monthly:

- Check `icon_backups` folder size
- Export important backups to external location

- Test restore operation to verify functionality
- Review and update cleanup limit if needed

Before Major Events:

- Windows Updates: Create manual backup with tag
- Hardware changes: Backup before and after
- System reinstall: Export all backups to safe location
- Monitor changes: Backup both before and after

11.6 Backup Archiving Strategy

Export Important Backups:

1. Navigate to `icon_backups` folder
2. Copy important .json files to safe location
3. Consider cloud storage (Dropbox, OneDrive, Google Drive)
4. USB drive backup for critical configurations
5. Include in regular system backup routine

When to Export:

- Perfect layouts you want to preserve forever
- Before system migrations or reinstalls
- Company/work standardized layouts
- Configurations that took significant time to create

12 Frequently Asked Questions (FAQ)

12.1 General Questions

Q: Does the program work with Windows 11?

A: Yes, fully compatible with Windows 11 and all versions back to Windows 7. The program uses standard Windows APIs that work across all versions.

Q: Can I use the program with virtual desktops?

A: Windows virtual desktops share the same physical desktop icon layout. Backups include all visible icons regardless of which virtual desktop is active.

Q: Are backups portable between different computers?

A: Yes, JSON files can be copied between computers. However, positions may not be accurate if:

- Screen resolutions differ (use adaptive scaling)
- Monitor configurations differ (different number or arrangement)
- Some icons in backup don't exist on target system

Q: How much disk space do backups use?

A: Each backup typically occupies 2-10 KB depending on icon count. With 50 backups, total usage is generally less than 500 KB (0.5 MB).

Q: Can I manually edit backup files?

A: Yes, they are standard JSON text files. You can edit with any text editor. Be careful with JSON syntax to avoid corruption. Always keep backup copies before editing.

12.2 Feature Questions

Q: Why create backup before scramble?

A: This mandatory backup ensures you can always restore your original layout. Without it, scrambling would be destructive with no undo option.

Q: What happens to icons not in backup during restore?

A: Icons currently on desktop but not in backup remain where they are. Only icons that exist in both the backup and current desktop are moved.

Q: Does adaptive scaling work with multi-monitor setups?

A: Scaling uses primary monitor resolution. For multi-monitor setups, it's recommended to create separate backups for each specific configuration rather than relying on scaling.

Q: Can I run multiple instances?

A: No, running multiple instances simultaneously is not supported and may cause conflicts. Only one instance should run at a time.

Q: Does the program auto-update?

A: Currently, no auto-update feature exists. Check the GitHub repository or project website for new releases and manual download.

12.3 Technical Questions

Q: Why does the program need Windows API access?

A: Desktop icon positions are managed by Windows Explorer's ListView control. The program must use Win32 API calls to read and modify these positions, as there's no higher-level interface available.

Q: Is administrator access required?

A: No, the program runs with standard user permissions. It only accesses your own user desktop, not system-wide or other user desktops.

Q: Can antivirus software interfere?

A: Some aggressive antivirus programs may flag the memory access operations as suspicious. Add the program to your antivirus whitelist if you encounter issues.

Q: Why monospaced font in Backup Manager?

A: Monospaced fonts (Consolas) ensure perfect column alignment without complex UI frameworks, making the backup list easy to scan and read.

Q: Can I backup icon arrangement to network drive?

A: The `icon_backups` folder must be local for proper operation. After creation, you can manually copy backups to network storage for archiving.

13 Technical Information

13.1 Software Architecture

Core Components:

- **PyQt6 Framework**: Modern cross-platform GUI framework providing widgets, layouts, and event handling
- **pywin32 Library**: Python bindings for Windows API, enables low-level desktop access
- **QThread**: Asynchronous operations prevent UI freezing during lengthy operations
- **QSettings**: INI-based configuration persistence across sessions
- **QSystemTrayIcon**: Native system tray integration with context menus

Key Classes:

- `DesktopIconManager`: Core logic for save/restore/scramble operations
- `IconWorker`: QThread worker for background processing
- `MainWindow`: Primary application window and UI orchestration
- `BackupManagerWindow`: Backup selection and management dialog
- `IconPreviewWidget`: Custom widget for visual layout preview

13.2 Remote Memory Access Process

To read icon positions, the program performs complex low-level operations:

Detailed Process:

1. **Find Desktop ListView**:
   - Locate ”Progman” window (Program Manager)
   - Find ”SHELLDLL_DefView” child window
   - Locate ”SysListView32” control (icon container)
   - Fallback: Enumerate all windows if not found in standard location
2. **Access Remote Process**:
   - Get Explorer process ID via `GetWindowThreadProcessId`
   - Open process with `PROCESS_ALL_ACCESS` rights
   - Allocate 4096 bytes in remote process memory
3. **Query Icon Data**:
   - Send `LVM_GETITEMCOUNT` message for icon count
   - For each icon: Send `LVM_GETITEMPOSITION` for coordinates
   - For each icon: Send `LVM_GETITEMTEXT` for icon name
   - Read data from remote memory to local process

4. **Cleanup**:
    - Free allocated remote memory with `VirtualFreeEx`
    - Close process handle properly

Constants Used:

```
LVM_GETITEMCOUNT = 0x1004     # Get total icon count
LVM_GETITEMTEXT = 0x102D      # Get icon text/name
LVM_GETITEMPOSITION = 0x1010  # Get icon X,Y position
LVM_SETITEMPOSITION = 0x100F  # Set icon X,Y position
MEM_COMMIT = 0x1000           # Commit memory allocation
MEM_RELEASE = 0x8000          # Release memory
PAGE_READWRITE = 0x04         # Read/Write permissions
```

13.3 Data Structures

LVITEM Structure (packed binary format):

```
Format: 'IIIIIQI' (struct.pack)
- mask (I): 0x0001 (LVIF_TEXT flag)
- iItem (I): Icon index
- iSubItem (I): 0 (main item)
- state (I): 0
- stateMask (I): 0
- pszText (Q): Pointer to text buffer in remote memory
- cchTextMax (I): 512 (max text length)
```

Position Data:

```
Format: Two 32-bit integers (8 bytes)
- X coordinate (signed int)
- Y coordinate (signed int)
Unpacked with: struct.unpack('ii', data)
```

13.4 Threading Architecture

Why Threading Is Essential:

- Icon scanning can take 2-5 seconds for many icons
- Without threading, UI would freeze completely
- User cannot cancel or see progress without threading
- Windows messages would not be processed during operations

Worker Thread Design:

- **log_signal**: Emits log messages back to main thread
- **progress_signal**: Emits 0-100 progress values
- **finished_signal**: Emits completion status and optional data
- Main thread updates UI in response to signals
- Worker thread performs all Win32 API calls

13.5 Desktop Refresh Mechanism

After modifying icon positions, the program forces desktop refresh:

```
# Re-enable window redrawing
win32gui.SendMessage(hwnd, WM_SETREDRAW, 1, 0)

# Invalidate entire ListView area
win32gui.InvalidateRect(hwnd, None, True)

# Broadcast system-wide setting change
win32api.SendMessage(
    HWND_BROADCAST,
    WM_SETTINGCHANGE,
    0,
    "IconMetrics"
)
```

Why This Is Necessary:

- Redrawing is disabled during restore for performance
- InvalidateRect forces visual update of moved icons
- WM_SETTINGCHANGE notifies all applications
- Without this, icons may not appear moved until manual refresh

13.6 Settings Storage

Configuration stored in INI format for human readability:

settings.ini Location:

```
[Application Directory]/settings.ini
```

Stored Settings:

```
[MainWindow]
geometry=@Rect(100 100 650 600)  # Window position and size

[Settings]
start_minimized=false
auto_save_on_exit=true
auto_restore_on_startup=false
adaptive_scaling_enabled=true
close_to_tray=false
cleanup_limit=10
```

13.7 Security and Privacy

Privacy Commitment

- No data is ever sent online or to external servers
- All backups are stored locally on your computer
- No telemetry, analytics, or usage tracking
- No network connections made by the program
- Access only to your own desktop (not other users)
- Source code is available for inspection

Data Stored:

- Icon names and positions only
- Screen resolution metadata
- User-provided descriptions/tags
- Application preferences (local settings.ini)

No Sensitive Data:

- No file contents or documents
- No passwords or credentials
- No browsing history or personal information
- No system information beyond screen resolution

14 Advanced Usage

14.1 Command Line Arguments

The application supports command-line arguments for automation and scripting. When any CLI argument is used, the program operates in a ”headless” mode, meaning the graphical user interface (GUI) will not be initialized, and the process will exit immediately after completion.

- **Silent Backup**: `--backup --silent` Performs a backup using the settings defined in `settings.ini` (e.g., respecting the maximum backup count).
- **Silent Restore (Latest)**: `--restore latest --silent`
  Automatically restores the most recent backup available in the backup directory.
- **Specify Backup File**: `--restore "filename.json" --silent`
  Restores desktop icon positions from a specific JSON file.

14.2 Technical Implementation Details

The CLI integration is designed to be fully compatible with the existing infrastructure:

- **Configuration Inheritance**: The CLI mode automatically loads the `settings.ini` file. This ensures that features like *Adaptive Scaling* and *Cleanup Limits* are applied even when running without a GUI.
- **Exit Codes**:
  - `0`: Operation completed successfully.

  - 1: An error occurred (e.g., file not found or permission denied).

14.3 Example Batch Script

You can create a .bat file to automate your backup on Windows startup or via Task Scheduler:

```
@echo off start "" "desktop-icon-backup-manager.exe" --backup --silent
```

14.4 Automation with Task Scheduler

You can create scheduled automatic backups using Windows Task Scheduler:

1. Open Task Scheduler
2. Create Basic Task
3. Name: "Desktop Icon Backup - Daily"
4. Trigger: Daily at preferred time
5. Action: Start a program
6. Program: Path to desktop-icon-backup-manager.exe
7. Enable: "Run whether user is logged on or not"

Important Notes:

- Program must be running for scheduled task
- Or enable "Auto-Save on Exit" and schedule shutdown after
- Task runs in background if "Start Minimized" is enabled

14.5 Batch Operations with JSON

For advanced users comfortable with JSON editing:

Merge Multiple Backups:

1. Open two backup JSON files
2. Copy icon entries from one "icons" section to another
3. Ensure no duplicate icon names (second will override first)
4. Save merged file with new name
5. Restore merged backup

Manual Position Adjustment:

1. Open backup JSON in text editor
2. Locate icon name in "icons" section
3. Modify [X, Y] coordinates (e.g., [100, 200])
4. Save file
5. Restore modified backup

Bulk Position Shift:

- Write script to add offset to all coordinates

- Useful for moving entire layout left/right/up/down
- Python script example available in project documentation

15 Development and Contribution

15.1 Building from Source

Requirements:

```
Python 3.8+
PyQt6
pywin32
```

Setup:

```
git clone [repository-url]
cd desktop-icon-backup-manager
pip install -r requirements.txt
python desktop-icon-backup-manager.py
```

Building Executable with PyInstaller:

```
pip install pyinstaller
pyinstaller --onefile --windowed --icon=icon.ico \
    --name="desktop-icon-backup-manager" \
    desktop-icon-backup-manager.py
```

15.2 Translation/Localization

The program uses Qt's translation system for internationalization:

Creating New Translation:

1. Extract translatable strings: `pylupdate6 *.py -ts i18n/lang.ts`
2. Translate strings using Qt Linguist
3. Compile: `lrelease i18n/lang.ts`
4. Place .qm file in i18n folder
5. Program auto-detects system language

Currently Supported Languages:

- English (en)
- Italian (it)
- Add your language via contribution!

16 Technical Information

16.1 Software Architecture

The program uses:

- **PyQt6**: Modern GUI framework
- **win32api**: Low-level access to Windows functions

- **Threading**: Asynchronous operations to avoid UI blocking
- **QSettings**: Configuration persistence in INI format

16.2 Remote Memory Access

To read icon positions, the program:

1. Locates the Explorer process managing the desktop
2. Allocates memory in the remote process
3. Uses Windows messages to query the ListView control
4. Reads coordinates and icon names
5. Frees allocated memory

16.3 Security and Privacy

- No data is sent online
- All backups are local
- No telemetry collection
- Access only to desktop information

17 License and Credits

17.1 Developer

mapi*68

17.2 Libraries Used

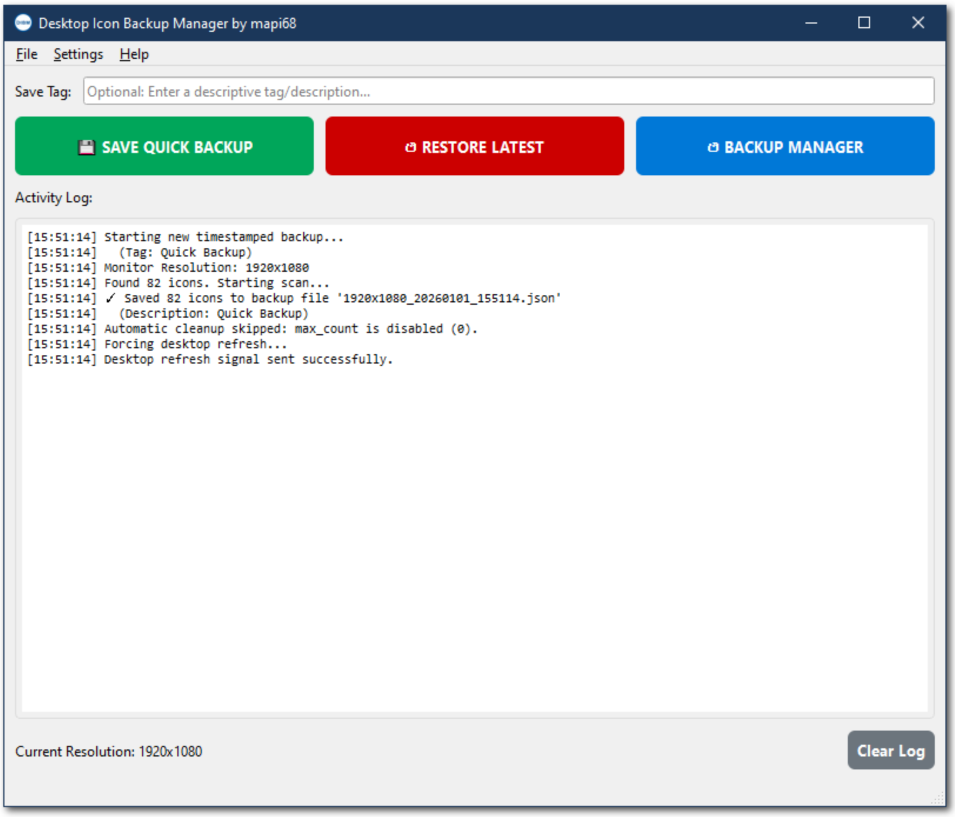
- PyQt6 - The Qt Company (GPL/Commercial License)
- pywin32 - Python for Windows Extensions (PSF License)

18 Support and Contacts

To report bugs, request features, or get support:

- Use the ”Issues” section of the repository
- Always include the complete error log
- Specify program version and Windows version

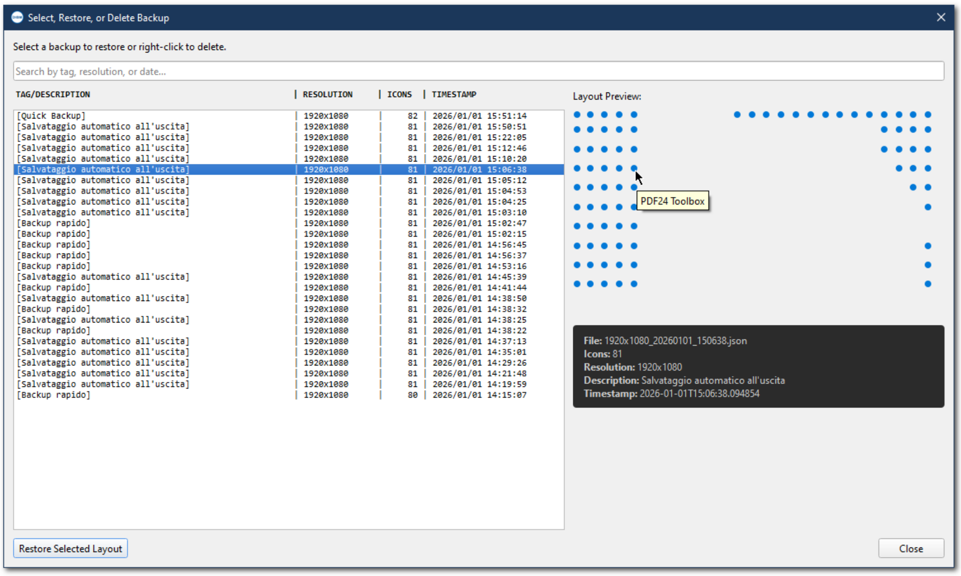
19 Screenshots



```
[15:51:14] Starting new timestamped backup...
[15:51:14]   (Tag: Quick Backup)
[15:51:14] Monitor Resolution: 1920x1080
[15:51:14] Found 82 icons. Starting scan...
[15:51:14] ✓ Saved 82 icons to backup file '1920x1080_20260101_155114.json'
[15:51:14]   (Description: Quick Backup)
[15:51:14] Automatic cleanup skipped: max_count is disabled (0).
[15:51:14] Forcing desktop refresh...
[15:51:14] Desktop refresh signal sent successfully.
```

Current Resolution: 1920x1080

Figure 1: Main interface showing the activity log and three main action buttons

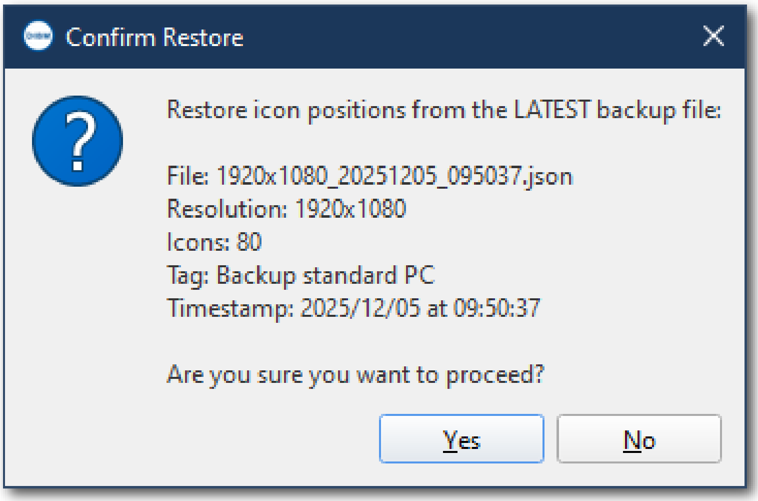

Select, Restore, or Delete Backup
Search by tag, resolution, or date...
Layout Preview:
PDF24 Toolbox
Restore Selected Layout
Close

Select a backup to restore or right-click to delete.

TAG/DESCRIPTION	RESOLUTION	ICONS	TIMESTAMP
[Quick Backup]	1920x1080	82	2026/01/01 15:51:14
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:50:51
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:22:05
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:12:46
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:10:20
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:06:38
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:05:12
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:04:53
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:04:25
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 15:03:10
[Backup rapido]	1920x1080	81	2026/01/01 15:02:47
[Backup rapido]	1920x1080	81	2026/01/01 15:02:15
[Backup rapido]	1920x1080	81	2026/01/01 14:56:45
[Backup rapido]	1920x1080	81	2026/01/01 14:56:37
[Backup rapido]	1920x1080	81	2026/01/01 14:53:16
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:45:39
[Backup rapido]	1920x1080	81	2026/01/01 14:41:44
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:38:50
[Backup rapido]	1920x1080	81	2026/01/01 14:38:32
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:38:25
[Backup rapido]	1920x1080	81	2026/01/01 14:38:22
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:37:13
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:35:01
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:29:26
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:21:48
[Salvataggio automatico all'uscita]	1920x1080	81	2026/01/01 14:19:59
[Backup rapido]	1920x1080	80	2026/01/01 14:15:07

File: 1920x1080_20260101_150638.json
Icons: 81
Resolution: 1920x1080
Description: Salvataggio automatico all'uscita
Timestamp: 2026-01-01T15:06:38.094854

Figure 2: Backup Manager window with list of saved backups and layout preview

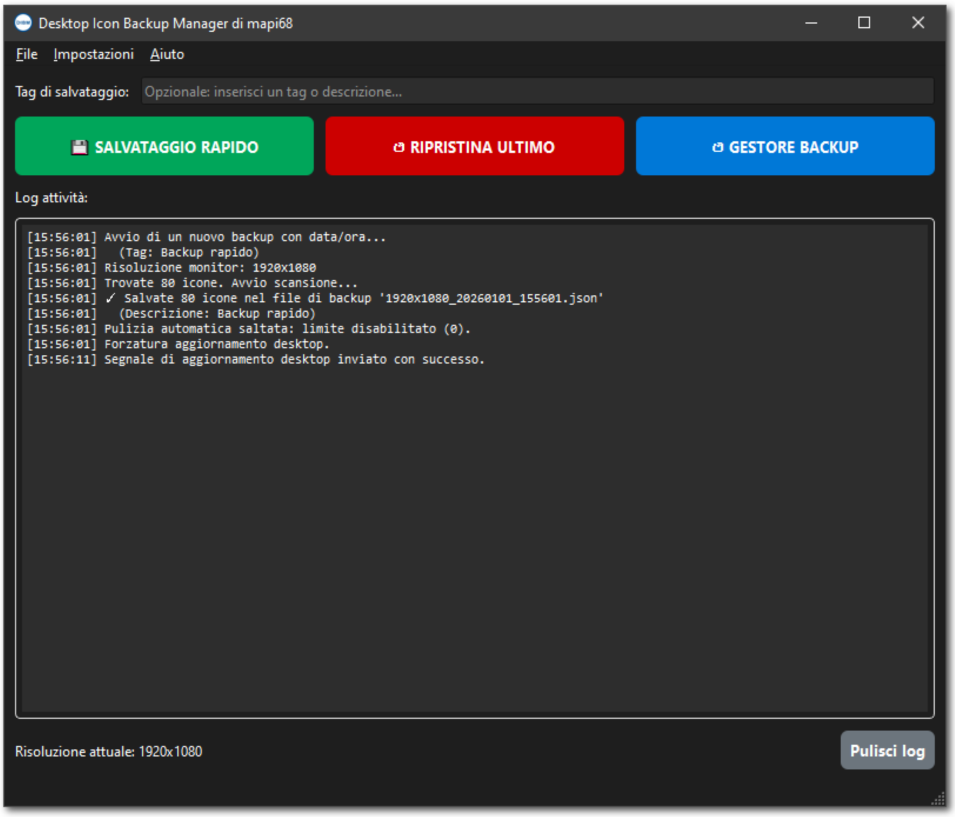


Restore icon positions from the LATEST backup file:

File: 1920x1080_20251205_095037.json
Resolution: 1920x1080
Icons: 80
Tag: Backup standard PC
Timestamp: 2025/12/05 at 09:50:37

Are you sure you want to proceed?

Figure 3: Confirmation dialog before restoring a backup

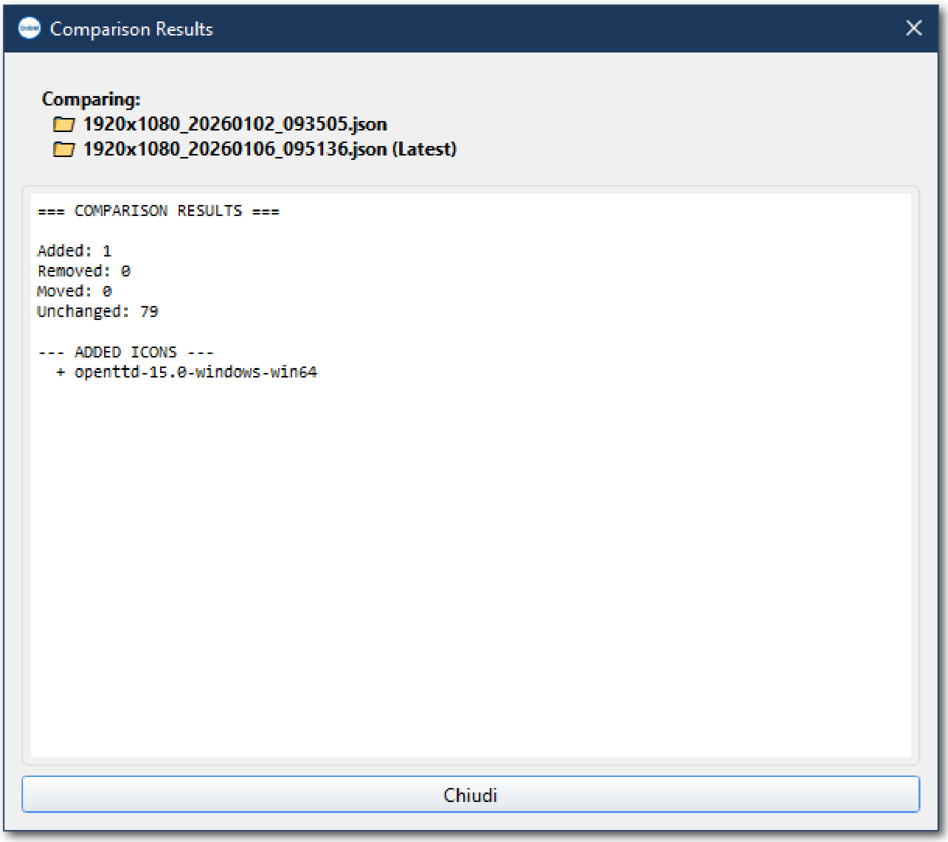


Tag di salvataggio: Opzionale: inserisci un tag o descrizione...

```
[15:56:01] Avvio di un nuovo backup con data/ora...
[15:56:01]   (Tag: Backup rapido)
[15:56:01] Risoluzione monitor: 1920x1080
[15:56:01] Trovate 80 icone. Avvio scansione...
[15:56:01] ✓ Salvate 80 icone nel file di backup '1920x1080_20260101_155601.json'
[15:56:01]   (Descrizione: Backup rapido)
[15:56:01] Pulizia automatica saltata: limite disabilitato (0).
[15:56:01] Forzatura aggiornamento desktop.
[15:56:11] Segnale di aggiornamento desktop inviato con successo.
```

Risoluzione attuale: 1920x1080

Figure 4: Desktop Icon Backup Manager featuring dark mode and Italian support



Comparing:
1920x1080_20260102_093505.json
1920x1080_20260106_095136.json (Latest)

```
=== COMPARISON RESULTS ===

Added: 1
Removed: 0
Moved: 0
Unchanged: 79

--- ADDED ICONS ---
  + openttd-15.0-windows-win64
```

Figure 5: Detailed view of the comparison interface

End of User Manual

Desktop Icon Backup Manager

January 6, 2026