Exchange Contacts Mirroring & Global Distribution Automation

Summary

This automation synchronizes contacts between licensed user mailboxes and a shared mailbox folder named **All Contacts** using Microsoft Graph app-only authentication.

It provides:

- One centrally maintained contact set.
- Automatic distribution back to users.
- Traceability in `personalNotes` with sync metadata.

Functional Overview

Step 1: User -> Shared mailbox

- Reads each user's default contacts (`/users/{id}/contacts`).
- Writes into shared mailbox folder **All Contacts**.
- Also stamps metadata back to source user contacts when needed.

Step 2: Shared mailbox -> User

- Reads shared mailbox contacts only from folder **All Contacts**.
- Syncs them into each user's default contacts.

Matching and deduplication

Contacts are matched in this order:

1. `SyncId` in `personalNotes`
2. Primary email (`emailAddresses[0].address`)
3. `displayName` (fallback)

Metadata Stamping (`personalNotes`)

The script maintains sync tags in `personalNotes`:

- `SyncId=<guid>`
- `CreatedBy=<upn-or-id>`
- `LastUpdatedAt=<utc-iso8601>`
- `LastUpdatedBy=<upn-or-id>`

Notes behavior:

- Existing non-sync note text is preserved.
- Old sync tags are replaced with fresh values.
- `SyncId` is generated if missing.
- `CreatedBy` is preserved from existing synced contact when available.

Update Behavior

`UpdateExisting` is enabled by default.

An existing target contact is only updated if:

- Data has changed, **and**
- Target contact is older than source (`lastModifiedDateTime` comparison).

If no change or target is newer/equal, it is skipped.

User Scope / Test Mode

User list resolution:

1. Per-tenant test users (if configured).
2. Otherwise all licensed users from Graph:
   - `/users?$filter=assignedLicenses/any(x:x/skuId ne null)`

Shared mailbox account is excluded from user sync loops.

Configuration

Preferred: full JSON config

Use `AUTOMATION_SYNCCONTACTS_CONFIG_JSON`:

```
{
  "MicrosoftTenants": [
    {
      "TenantId": "tenant-id-or-domain",
      "ClientId": "app-client-id",
      "ClientSecret": "app-client-secret",
      "GlobalAddressBookUserId": "shared.contacts@contoso.com",
      "TestUserList": ["user1@contoso.com"]
    }
  ],
  "DryRun": "true"
}
```

Multi-tenant via separate env var

Use `AUTOMATION_SYNCCONTACTS_MICROSOFT_TENANTS_JSON` with tenant array.

Backward-compatible single-tenant env vars

- `AUTOMATION_SYNCCONTACTS_TENANT_ID`
- `AUTOMATION_SYNCCONTACTS_CLIENT_ID`
- `AUTOMATION_SYNCCONTACTS_CLIENT_SECRET`
- `AUTOMATION_SYNCCONTACTS_GLOBAL_ADDRESS_BOOK_USER_ID`
- `AUTOMATION_SYNCCONTACTS_DRY_RUN` (`true/false`)

Optional tenant test-user map env var

Works even with full config JSON:

- `AUTOMATION_SYNCCONTACTS_TEST_USER_LIST_BY_TENANT_JSON`

Example:

```
{
  "contoso.onmicrosoft.com": ["user1@contoso.com", "user2@contoso.com"],
  "fabrikam.onmicrosoft.com": ["user3@fabrikam.com"]
}
```

Lookup keys include tenant identifiers like `TenantId`, `PrimaryDomain`, `TenantDomain`.

Prerequisites

- Shared mailbox exists and is reachable as `GlobalAddressBookUserId`.
- Azure AD app registration with application permissions.
- Admin consent granted.
- App-only access to mailbox contacts.

Required Microsoft Graph Permissions

Application permissions:

- `Contacts.ReadWrite`
- `User.Read.All` or `Directory.Read.All`

Dry Run

Set `DryRun` / `AUTOMATION_SYNCCONTACTS_DRY_RUN` to `true` to simulate actions without writes. The script logs what would be created/updated and prints creation summaries.

CI/CD

Designed for automation pipelines (for example GitLab CI/CD). Store secrets as protected CI/CD variables.

Troubleshooting

- **No tenants configured**: set `AUTOMATION_SYNCCONTACTS_CONFIG_JSON` or `AUTOMATION_SYNCCONTACTS_MICROSOFT_TENANTS_JSON`.
- **Insufficient privileges**: missing admin consent or Graph permissions.
- **Shared mailbox not found**: incorrect `GlobalAddressBookUserId`.
- **No licensed users found**: tenant has no users matching license filter.
- **Config JSON parse errors**: invalid JSON in env vars.

Security Notes

- Keep `ClientSecret` in secure secret storage.

- Restrict app permissions to minimum required.
- Limit and audit access to shared mailbox contact data.

Script Source

Automation-ContactsMirroring.ps1

Revision #4
Created 2026-01-09 21:50:07 UTC by Luca Noah Caprez
Updated 2026-02-24 16:22:56 UTC by Luca Noah Caprez