

How It Works

The Dataverse Mail Client is a React-based PCF virtual control that communicates entirely through the Dataverse Web API. Here's how key features operate.

Sending Emails

When you compose and send an email, the control performs three Web API operations in sequence:

1. **Create the email record** with direction code set to outgoing, activity parties (To, CC, From), subject, HTML body, and optional Regarding reference.
2. **Create attachment records** for each file — stored as `activitymimeattachment` records linked to the email.
3. **Execute the SendEmail action** with `IssueSend: true` — this hands the email to server-side sync for delivery through Exchange.

Receiving Emails

The control does not poll for new emails itself. Instead, it relies on Dataverse server-side synchronization:

- Exchange sync brings emails into the `email` table automatically.
- When you open the control or click **Refresh**, it queries the `email` table for recent records owned by you (personal) or associated with your queues (shared).
- Emails are fetched with their activity parties, allowing the control to display sender and recipient information.

Conversation Threading

Emails are grouped into conversations using two criteria:

- **Normalized subject** — the subject line with RE:, FW:, and similar prefixes stripped.
- **Regarding record** — if set, emails about the same record are grouped together even if subjects differ slightly.

Within a thread, messages are displayed chronologically. You can expand/collapse individual messages to focus on what matters.

Personal vs Queue Mailboxes

The control separates emails into two categories:

- **Personal mailbox:** Emails where you are the owner (`_ownerid_value` matches your user ID). Queue emails are excluded to avoid duplicates.
- **Queue mailbox:** Emails associated with a queue through `queueitem` records. The control first fetches queue item IDs, then retrieves the corresponding email records.

Recipient Search

When you type in the To or CC fields, the control searches across three tables simultaneously:

TABLE	FIELDS SEARCHED
Contact	<code>fullname</code> , <code>emailaddress1</code>
Account	<code>name</code> , <code>emailaddress1</code>
System User	<code>fullname</code> , <code>internalemailaddress</code>

Results are combined and displayed as a type-ahead dropdown, showing the entity type, name, and email address for each match.

Entity Filtering

The inbox supports filtering by specific records. When you apply a record filter:

- The control queries emails where the `_regardingobjectid_value` matches the selected record.
- This lets you see all email communication related to a specific Account, Contact, Case, or any other entity — regardless of which mailbox they arrived in.