

Run your tests against maintained provider behavior.

Telryn is in development. We're building test environments to reduce mock upkeep and repeated provider investigation. Your team would run maintained cases alongside its existing integration tests, with supporting evidence and reviewed corrections.

No product access yet. SharePoint is the first development focus; launch timing and commercial terms have not been set.

Start with one case your team needs to test.

1. **Prepare.** Agree on the case, provider evidence and gaps. Pin versions and supply minimized fixtures. Configure approved test endpoints and authentication.
2. **Run.** Your tests run the connector and check each phase. Telryn records provider observations and checks selected-condition coverage separately.
3. **Repeat.** Keep state within a multi-phase case. Reset provider and application state before independent runs. Your team checks fixes and controls release.
4. **Review updates.** Review Telryn's findings and corrections, then rerun your checks before adopting a new case or environment version.

Hypothetical example, not product results

Fetch, add and update before testing recovery.

Provider conditions and customer-owned application checks

Phase	Provider condition	Your application check
Initial fetch	Serve the starting documents and a change checkpoint.	Expected document IDs and contents are present.
Addition	Add a declared file; the connector polls for changes.	The new document appears with expected content.
Update	Change an existing file's content while retaining its ID.	Content updates without another stored document.

A later, separately selected case could throttle a page request and check recovery. Basic sync coverage does not establish fault handling or all permission behavior.

First development focus: SharePoint document sync.

We are qualifying initial fetch, additions and same-ID content updates through Microsoft Graph polling delta, then one selected error case. This is not released provider support. Authentication, deployment compatibility and wider provider fidelity need separate qualification. Additional providers have not been selected.

Keep the case useful after the first run.

Telryn would investigate discrepancies and maintain runnable cases with reviewed corrections. Each update would include sources and dates, affected conditions, a pinned version and known gaps. Your team would own application code, data, checks, fixes and releases. A local provider container and CLI are the first delivery mechanism in development.

Setup, endpoint configuration and review take customer time. Any engagement would need an agreed scope, permitted evidence, price and follow-up. Subsequent use needs to remove enough investigation and upkeep to justify continued maintenance; those savings remain unproven.

Check the environment as well as the application.

Telryn would check its provider behavior against documentation and authorized observations. Its run check would establish only whether selected provider conditions were exercised. Your tests would judge application behavior. A pinned case would retain its evidence date and limits, not certify current provider behavior.

Repeatable provider conditions do not make an entire application deterministic or prove the cause of a past incident. Existing mocks, simulators or sandboxes may already cover your case at an acceptable maintenance cost.