



Temporal at Groww



Story before temporal

- Normal scenario - Requires an API call to an “unreliable” third party

User id	Critical status
123	PENDING/SUCCESSFUL

- Unreliable third party data store

Userid	API status
123	FAILED_TIMEOUT



Why temporal

- The second table is implicit
- Removes overhead of managing the second table (periodic crons to retry failed)
- Removes overhead of maintaining consistency across the two tables



What temporal brings to Groww - contd

- Alternatives/ Workflow as a unit of scale
- Workflow as code (oops concepts can be applied quickly to distributed systems) - testing, activities as interfaces, versioning, idempotency, workflow lifecycle definitions



Looking forward

- Improve our monitoring of temporal/workers
- Build standards around workflow config which are defensive, safe and reduce unpredictability
- Choices of SQL vs Cassandra