Text stack

The text stack follows the spatial design with one repository per role, so documents become tables an agent queries like any other layer.

	Spatial stack — IN PRODUCTION		Text stack — NEW
RECIPES	**geo-workflows** One reviewed recipe per dataset, from issue to published collection.	→	**text-workflows** PILOT One recipe per corpus, setting its sources, scope, languages and gazetteers.
BUILDER	**geo-ingest** Converts vector and raster sources and indexes them on the H3 grid, as parallel Jobs.	→	**text-ingest** IN DESIGN Turns documents into tables of passages and references, with OCR, structure and places.
DATA ON S3	`GeoParquet` `PMTiles` `COG` `H3 hex parquet`	→	`documents` `passages` `references` `H3 places`
METADATA	A STAC catalog that the apps and the agent both read.	→	Croissant and CSL-JSON, with a thin STAC record so the catalog can find the corpus.
QUERY TOOL	**geo-query** An MCP server for SQL over hex parquet, guided by STAC.	→	**text-query** LIVE IN ONE APP An MCP server that runs codebooks over any text column a SQL query returns.

Running today in the private High Seas app

The app's agent codes passages from a literature corpus that was ingested by hand. The corpus sits in a private rustfs bucket, read with a key scoped to it.

`high-seas-private` `high-seas-text-query` `high-seas-lit`

One grid for both

Places named in a passage are stored as H3 cells, so coded passages join the spatial layers on the same grid.