Territory Building with AI

MCP Server · https://www.easyterritory.com/documentation/ezt-mcp/territory-building-with-ai/

Territory building is the core workflow of the MCP Server. You describe what you need in natural language, and the AI agent translates your request into the right combination of build tools. This guide covers the build approaches, the five questions that shape a balanced build, and worked examples you can try immediately.

Build Approaches

The server offers several build tools. Your agent chooses from what you asked for. These are not interchangeable.

Tool	Best For	How It Works
`auto_build`	Most situations — balanced territories from scratch	Groups nearby accounts into compact territories that meet your balance target. The recommended starting point.
`account_build`	Grouping accounts by an existing CRM field (e.g., rep name, region)	Builds territories that mirror an attribute already on your accounts — no optimization, just geometry.
`direct_build`	Importing an existing territory assignment file	Loads a pre-made assignment file straight onto the map. Use when you already have boundaries and just need to visualize or analyze them.
`cluster_points`	Balanced groups on the point layer — no territory polygons	Adds a balanced group-id column to accounts using the same sizing, objective, dwell, and visit-frequency controls as auto_build. Colors the groups on the map but does not create a territory layer or dissolve geography.

Tool	Best For	How It Works
`seed_build`	One territory grown from one location	Grows outward from a seed address or coordinate until a location count or a metric total is met as closely as possible. One territory per call. It does not divide the whole set, and it will not overlap an existing territory on the same layer.
`isochrone_build`	How far you can drive from a location	Draws a drive-time or drive-distance area. That is a reachability budget, not a balanced territory.

Use `auto_build` to divide accounts into compact, balanced territories. Use `seed_build` when you want one territory around one site — a franchise zone, a new branch, a protected area of about N locations. Use `cluster_points` for the same balance engine with only a grouping column on points. Use `account_build` or `direct_build` when the assignment already exists. Use `isochrone_build` only for drive-time reach, not for balance.

The Five Build Questions

Every `auto_build` and `cluster_points` call is shaped by five parameters. Your agent should confirm them before building. Do not invent dwell time — if there is no dwell column, give an average onsite time.

1. Sizing: Count or Workload Target?

Do you want a fixed number of territories (e.g., “build 5 territories”) or a target workload per territory (e.g., “build territories with roughly 200 accounts each”)? The former gives you control over the count; the latter lets the server decide how many territories are needed.

2. Balance Dimension

What metric should the server balance across territories? Options include account count, total revenue, total workload (visits × duration), or a custom numeric field on your accounts. For location-count balance, use a metric bias of 100 on the account count column with a workload bias of 0. Nothing is assumed: a build called without a balance objective stops and asks.

3. Bias

Bias controls the trade-off between drive-time workload and an output measure. A workload bias of 100 balances on workload (drive time plus dwell). A metric bias of 100 balances on a column such as revenue or account count. A blend uses both, and the two biases sum to 100. There is no default: a build called without a balance objective stops and asks.

4. Visit Frequency

If your accounts have different visit cadences (e.g., key accounts visited weekly, others monthly), you can specify a visit frequency field. The server factors this into the workload calculation so that high-frequency accounts contribute more to the balance target.

5. Dwell Time

Dwell time is the average time spent at each account per visit (in minutes). Combined with visit frequency and drive time between stops, this gives the server a realistic workload estimate. There is no silent default — the agent should ask if your data has no dwell column.

Scoping to a State or Region

Most builds are scoped to a specific geography. The server uses two parameters to filter the parts (ZIP codes, counties, or other geographic units) included in the build:

```
part_scope: "explicit"
part_filter: { "state_abbr": "TX" }
```

This restricts the build to ZIP codes in Texas. You can also filter by region, custom territories, or any geographic attribute the server recognizes. If you omit scoping, the server defaults to the parts that intersect the bounding box of your accounts, which can reach national scale. Set an explicit scope with a part filter to hold a build to one geography.

Territories can be built from ZIP codes, counties, Canadian FSAs, and international postal areas, including French 5-digit postal codes. Ask the agent to list the available layers if you're not sure which to use.

Supported Countries & Boundary Layers

These are the countries and boundary layers currently available for building territories. Counts are approximate.

- **United States:** ZIP codes (~33,700), counties (~3,200)
- **Canada:** FSAs, the first three characters of a postal code (~1,660)
- **United Kingdom:** postcode areas (125), districts (~2,880), sectors (~9,690)
- **Germany:** postal codes (~8,200)
- **Spain:** postal codes, including the Canary Islands (~10,860)
- **France:** 5-digit postal codes (~6,040)
- **Italy:** communes (~7,960), provinces (107), regions (20), macroregions (5)
- **Netherlands:** 4-digit postal codes (~4,010)
- **Norway:** postal codes (~3,270)
- **Switzerland:** postal codes (~4,090)
- **Belgium:** municipalities (~590)
- **Denmark:** postal codes (~590)

Ask the agent to list the available layers for the latest set.

Example: Balanced Territories in Texas

Say this to your agent:

```
“Build 5 balanced territories from my accounts in Texas, workload-only, 30-minute dwell time.”
```

The agent will:

1. Filter accounts to Texas (part_scope=explicit, part_filter with state_abbr=TX)
2. Call `auto_build` with 5 territories, balance dimension = workload, dwell = 30 minutes
3. Plot the resulting territories on the map, color-coded by territory
4. Report the balance metrics so you can see how even the workloads are

If the balance is off, you can ask the agent to rebuild with a different bias or adjust the territory count. Iteration is fast — each rebuild takes seconds.

Example: One Territory Around a Location

Say this when you want a single territory grown from a site, not a full partition:

```
"Build a territory around 123 Main Street with about 40 of my locations. Do not overlap the territories already on the map."
```

The agent geocodes the address, then calls `seed_build` once. The territory grows over adjacent ZIP codes or counties until the location count or metric total is as close as possible. A second site is a second call. This is not a drive-time ring — use `isochrone_build` for that.

Example: Mirror Existing Rep Assignments

If your accounts already have a rep assignment field (e.g., `assigned_rep` in your CRM), you can build territories that match:

```
“Build territories that mirror my rep assignments. Group accounts by the assigned_rep field.”
```

The agent will call `account_build` with the `assigned_rep` field as the grouping key. Each rep becomes a territory. This is useful when you want to visualize existing assignments and analyze their balance without changing them.

After Building: Analyze and Load Results

Building territories is only the first half of the workflow. Once territories exist on the map, ask the agent to analyze them:

```
“Analyze the current territory assignment and load the results panel.”
```

The agent calls `analyze` to compute balance metrics, drive-time statistics, and compactness scores, then `load_analysis_panel` to display them in an interactive panel on the map. You can compare territories side by side and identify outliers that need adjustment.

You can also see which accounts fall inside a territory or drive-time area.

Next Steps

- Territory Management & Realignment — Split, merge, rebalance, and fine-tune existing territories.
- Getting Started with EasyTerritory MCP — High-level overview of the full workflow.
- MCP Server Tools Reference — Full list of all 58 tools.