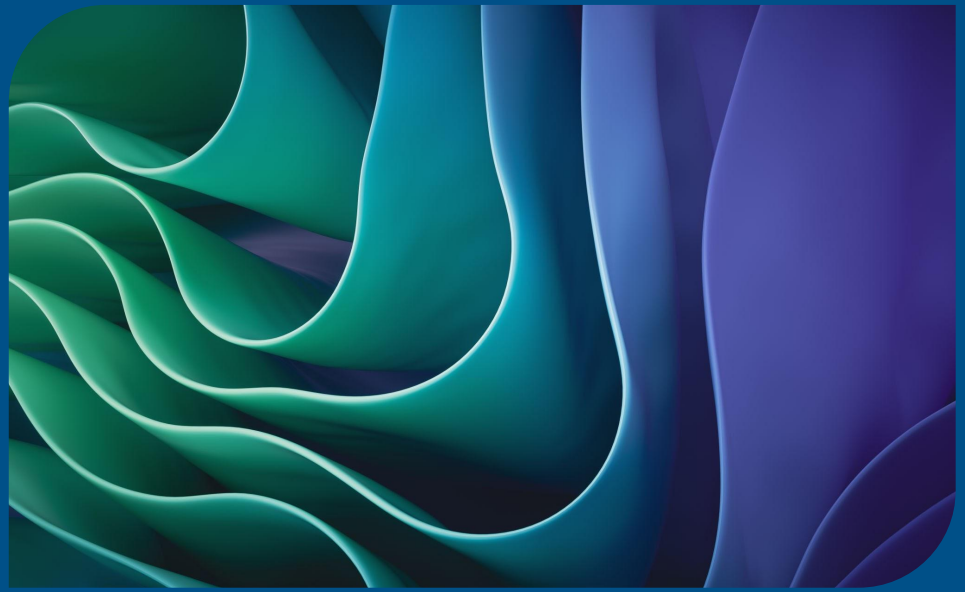
May 16, 2026

Project Development for Amateur Radio: You Can Do It (Really)

Rus Healy, K2UA



Dayton HamVention 2026

1. Our Opportunity
2. Prompt Engineering
3. Model Context Protocol
4. Use Your Imagination
5. Getting Started and Sharing with the World

1. **Our Opportunity**
2. Prompt Engineering
3. Model Context Protocol
4. Using Your Imagination
5. Getting Started and Sharing with the World

Have you ever had a really good idea, but didn’t have the skills to bring it to life?

AI can unlock *all of that* for you.

AI is polarizing, but for all the wrong reasons.

People fear the unknown.

We don’t have to be those people.

We are hams. We are builders.

AI gives us a *massive* new toolset to use for bringing our ideas to life.

Fundamentals

Large Language Models

- AI systems trained to understand and generate human-like text
- They learn from vast amounts of text—books, websites, and documents
- Use patterns learned during training to produce meaningful responses
- Work best when guided by clear instructions and examples

Prompts

- A way to guide the model toward useful, accurate, and creative outcomes
- Focus on clarity, context, and intent in your request
- Saves time and improves the quality of results in writing, analysis, or planning tasks
- Lets you *automate tasks, explore ideas, and solve real problems* efficiently

Model Context

- A standard way to connect AI models with data and tools
- Makes AI more reliable and grounded in real information
- Interoperability: Connects multiple systems without custom coding
- Security: Clear control over what information AI can see or use
- Models use this context to answer questions and perform actions more accurately

1. The Opportunity
2. **Prompt Engineering**
3. Model Context Protocol
4. Using Your Imagination
5. Getting Started and Sharing with the World

Prompt Engineering

"The new Google searching"

The most important way to spend time improving your generative AI results

Concept	Search Engine Skills	Prompt Engineering
Goal	Retrieve existing information	Generate entirely new content, code, or reasoning
Approach	Focus on fragmented keywords	Use full sentences, specific constraints, and guided instructions
Feedback	Static list of results	Conversational replies
Required Thinking	"What do I want to *find*?"	"What do I want to *do* or *create?*"
Example	"Best hiking trails in Utah"	"Create a three-day hiking itinerary for Utah with scenic routes and easy terrain."

Prompt Engineering

Effective tasking through detailed prompting

Task Goal	Weak Prompt	Strong Prompt
Write a program	“Write a Python script.”	“Write a Python script that asks for a user’s name, saves it to a text file, and prints a friendly greeting. Include clear comments.”
Debug code	“Fix this code.”	“Review this Python function that returns the wrong total for my contest log. Explain what causes the bug and show a corrected version with a short explanation.”
Generate documentation	“Add comments.”	“Add helpful inline comments to this Python function so a beginner can understand each step.”
Plan a project	“Build a web app.”	“Outline the key steps to build a simple web app that tracks household expenses. Include language options, database choice, and rough timeline.”
Suggest improvements	“Improve my app.”	“Suggest three improvements to make this expense-tracking app easier for seniors to use. Focus on interface design and clarity. Ask up to three clarifying questions before you begin.”

Prompt Engineering

Key Tips for the most important skill to learn when you're starting out with generative AI

- Be specific about the results you want–the format (text, table, etc), the length, and all other important details.
- For developing projects that include code, specify the language(s) you want the model to create (ie, Python).
- Ask the model to ask you clarifying questions before beginning! This ensures the best outcome in the least time.
- If you want to create a GitHub project, even for a single document, specify that. (GitHub is free and it makes it easy to share projects, to revise them later, and to collaborate with others on them.)

Good overview video from Anthropic on prompt engineering: https://www.youtube.com/watch?v=ysPbXH0LpIE

1. Our Opportunity
2. Prompt Engineering
3. **Model Context Protocol**
4. Using Your Imagination
5. Getting Started and Sharing with the World

Model Context Protocol

Giving the LLM specialized access and knowledge

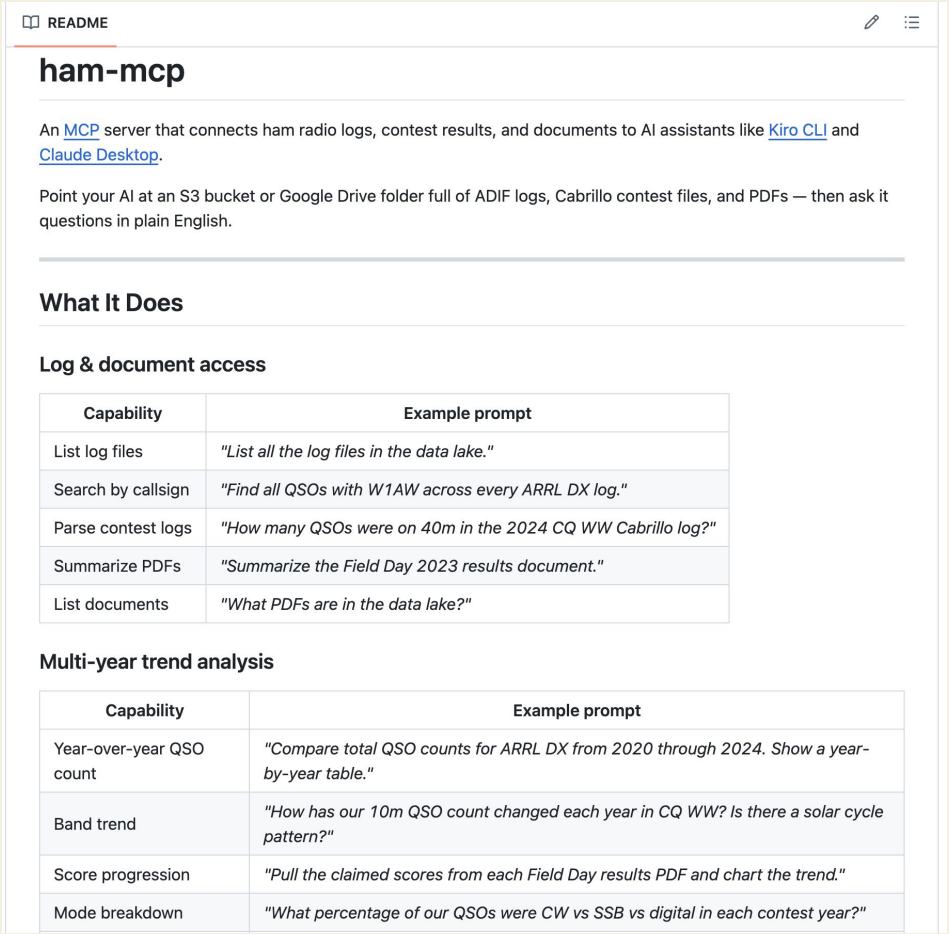
MCP Server	Purpose	Why It's Useful for Amateur Radio
Filesystem	Access and manage local files and directories	Essential for reading/writing configuration files, log files, and working with **radio control software** files
Sequential Thinking	Extended reasoning and problem-solving	Helps break down c**omplex RF propagation calculations, antenna design problems, and EME path loss analysis**
Memory	Store and retrieve information across sessions	Keep track of s**atellite passes,** QSO details, station configurations, and project progress
Brave Search	Web search capabilities	**Research propagation reports, look up callsigns, find technical specifications for radios and antennas**
Fetch	Retrieve web page content	**Pull real-time propagation data, check band conditions,** access repeater databases and other Amateur Radio resources
GitHub	Repository management	Store and version control your software projects, access existing Amateur Radio code repositories
SQLite	Database operations	**Manage QSO logs,** contact databases, repeater frequencies, and **EME scheduling information**
Time	Current time and timezone info	**Critical for EME scheduling, satellite tracking, contest logging with accurate UTC timestamps**
Google Maps	Location and mapping data	**Calculate beam headings, distances for grid square calculations, and visualize coverage areas**
Everything	Access to multiple integrated tools	Combines file, web, and data operations for comprehensive development workflow

Model Context Protocol

A real-world example:

https://github.com/rusk2ua/ham-mcp

You can run this MCP locally, or deploy it on AWS and use it to analyze logs and contest results with a private repository. Feel free to fork this repository and develop your own version of it!



README

ham-mcp

An MCP server that connects ham radio logs, contest results, and documents to AI assistants like Kiro CLI and Claude Desktop.

Point your AI at an S3 bucket or Google Drive folder full of ADIF logs, Cabrillo contest files, and PDFs — then ask it questions in plain English.

What It Does

Log & document access

Capability	Example prompt
List log files	*"List all the log files in the data lake."*
Search by callsign	*"Find all QSOs with W1AW across every ARRL DX log."*
Parse contest logs	*"How many QSOs were on 40m in the 2024 CQ WW Cabrillo log?"*
Summarize PDFs	*"Summarize the Field Day 2023 results document."*
List documents	*"What PDFs are in the data lake?"*

Multi-year trend analysis

Capability	Example prompt
Year-over-year QSO count	*"Compare total QSO counts for ARRL DX from 2020 through 2024. Show a year-by-year table."*
Band trend	*"How has our 10m QSO count changed each year in CQ WW? Is there a solar cycle pattern?"*
Score progression	*"Pull the claimed scores from each Field Day results PDF and chart the trend."*
Mode breakdown	*"What percentage of our QSOs were CW vs SSB vs digital in each contest year?"*

Part 2

Real Radio Projects

1. Our Opportunity
2. Prompt Engineering
3. Model Context Protocol
4. **Using Your Imagination**
5. Getting Started and Sharing with the World

Getting Out of Your Lane

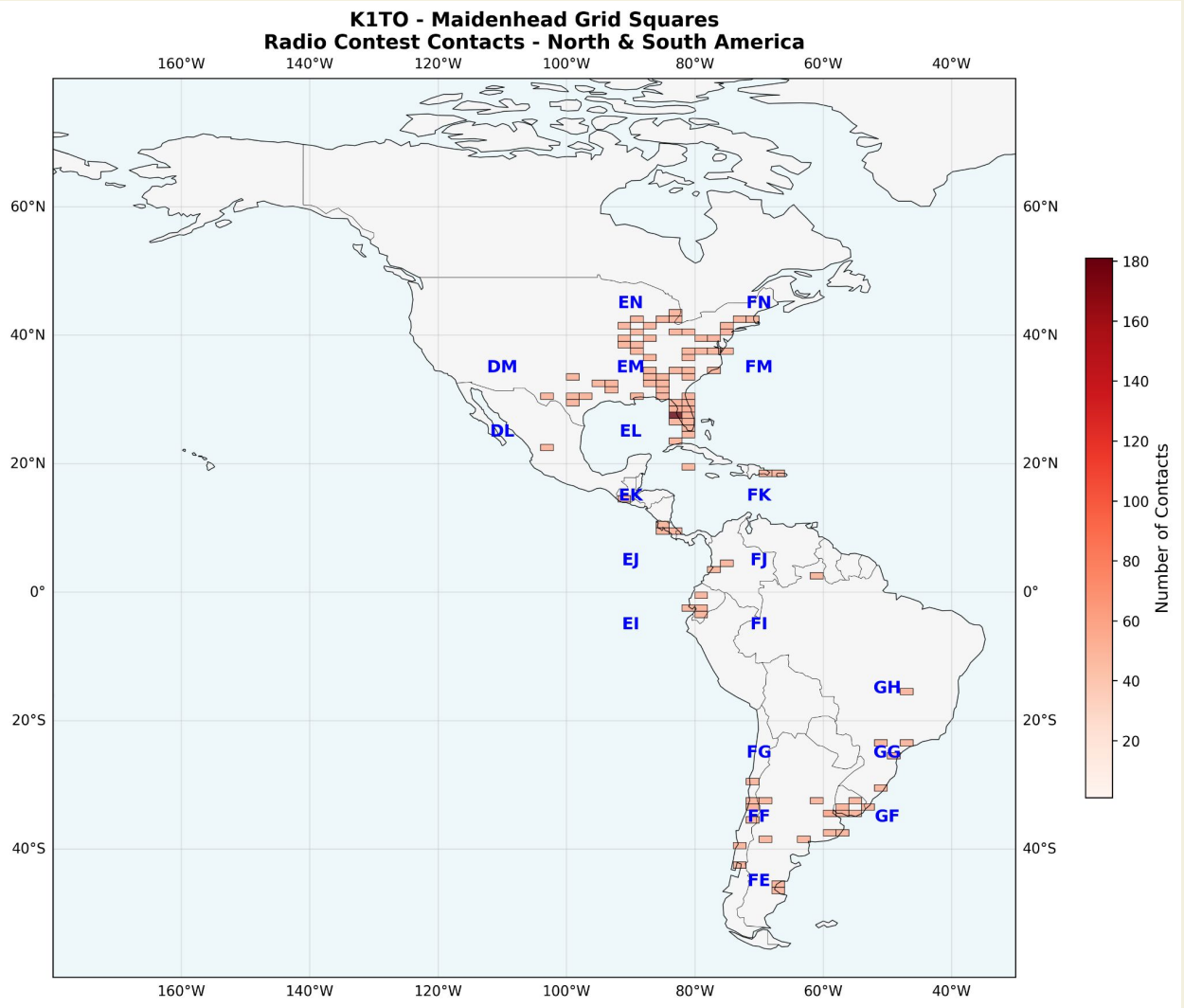
What could you do if you had no boundaries?

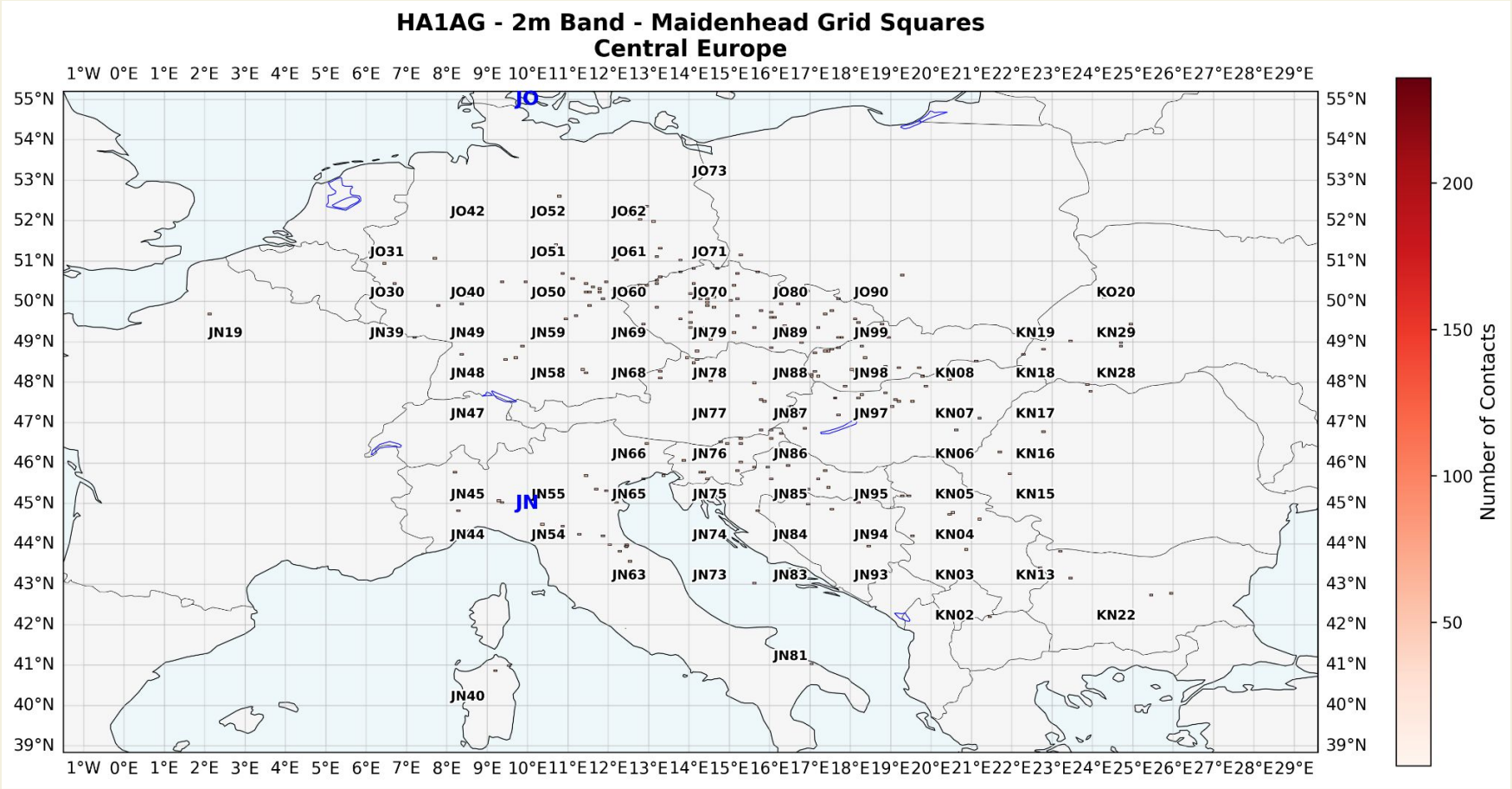
How would you use data to tell stories?

What could you do for the community?

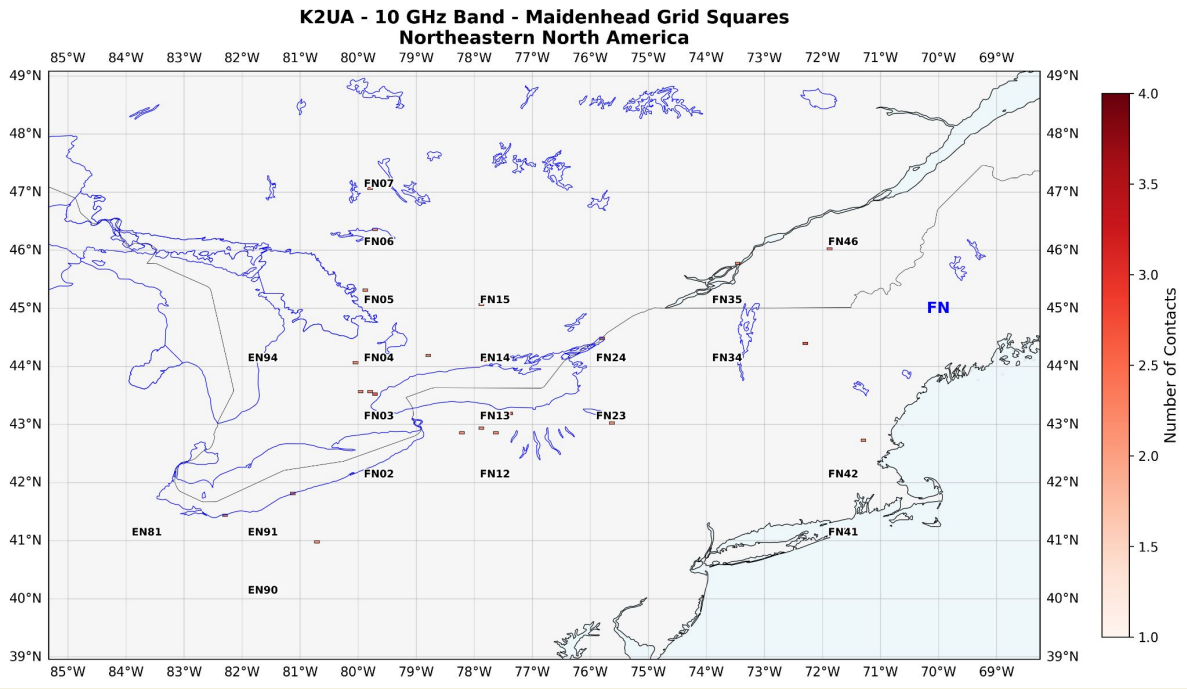
URL	Resource Type	Description
github.com/rushealy-aws/10ghz-logger-analyzer	**Contest Score Report**	Amateur radio contest claimed score submission with operator comments, equipment details, and operating experiences. Provides reference for performance analysis and learning strategies from experienced contesters.
github.com/rusk2ua/eme-calculator-project	**EME Planning Tool**	Calculator for optimal Earth-Moon-Earth dish placement analyzing moon position, azimuth ranges, wind loading, and tree obstructions across bands (144 MHz to 10+ GHz). Provides annual pass counts for target regions with web interface and AWS Lambda deployment.
github.com/rushealy-aws/grid-mapper	**QSO Visualization Tool**	Creates color-coded Maidenhead grid square maps from contest logs (Cabrillo/CSV) showing geographic coverage and contact density per band. Supports all amateur bands from 630m through 2mm with automatic continent detection.
github.com/rushealy-aws/10ghz-logger-analyzer	**Microwave Contest Logger**	Converts Google Sheets logs to Cabrillo format for ARRL 10 GHz contest with distance-based scoring and band multipliers (10 GHz-300+ GHz). Generates station reports, weekend analysis, and multi-log comparisons with bearing/propagation analysis.

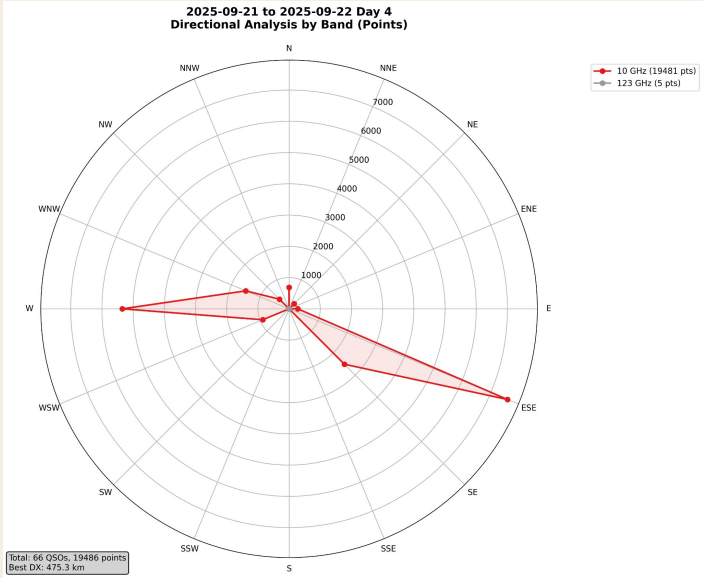
K1TO Sep VHF Maidenhead Field Map from Cabrillo file





https://github.com/rushealy-aws/grid-mapper





https://github.com/rushealy-aws/10ghz-logger-analyzer

EME Dish Siting Calculator

A comprehensive tool for calculating optimal Earth-Moon-Earth (EME) dish placement based on location, frequency band, and environmental factors.

Overview

This calculator helps amateur radio operators determine the best location on their property for EME dish installations by analyzing:

- Moon position calculations throughout the year
- Optimal azimuth ranges for target regions
- Wind loading considerations
- Tree/obstruction analysis
- Frequency-specific RF considerations

Features

- **Location Input**: Maidenhead grid square or lat/lon coordinates
- **Multi-band Support**: 144 MHz, 432 MHz, 1296 MHz, 2304 MHz, 3456 MHz, 5760 MHz, 10 GHz+
- **Target Regions**: Europe, Caribbean, South America, Africa, Asia, Oceania
- **Environmental Factors**: Tree heights, wind speeds, property boundaries
- **Operating Schedule**: Moonrise-based operating windows
- **Web Interface**: Easy-to-use calculator with visual results
- **Serverless Deployment**: AWS Lambda-based backend

https://github.com/rusk2ua/eme-calculator-project

Quick Start

Web Interface

Visit the deployed calculator at: `https://your-api-gateway-url.amazonaws.com`

Local Development

```
git clone https://github.com/yourusername/eme-dish-calculator.git
cd eme-dish-calculator
pip install -r requirements.txt
python src/eme_calculator.py --grid FN12fr46 --band 1296
```

Example Case Study: FN12fr46 on 23cm

Location: FN12fr46 (42.73°N, 77.55°W, 500m ASL)
Band: 1296 MHz (23cm)
Constraints: 80-foot trees west of house, 30-50 mph winds

Recommendation: Place 2.4m dish 150-200 feet east of house

- **Coverage**: 30°-210° azimuth range
- **Annual Opportunities**:
  - Europe: 240 passes

1. The Opportunity
2. Prompt Engineering
3. Model Context Protocol
4. Using Your Imagination
5. **Getting Started and Sharing with the World**

Getting Started and Sharing with the World

Sharing what you've built:

Create a free GitHub account

Tell your AI companion to package up your project for GitHub as part of your prompt

Bonus: Configure the GitHub MCP and it will handle everything for you securely.

Tools

Before you begin: *Watch these videos:*
https://www.youtube.com/watch?v=WSPChlfxJyA and
https://www.youtube.com/watch?v=DsKZpgoy830

Zero setup:

- Claude https://claude.ai and the Claude installable app ← best overall
- Google NotebookLM https://notebooklm.google.com ← easiest to use, powerful set of tools
- https://app.kiro.dev (Amazon's web-based tool)
- Standalone AI browsers and dev tools like **Comet** https://www.perplexity.ai/download-comet

→ Note that I haven't mentioned ChatGPT. I don't recommend it.

Some setup:

Development environments like:

- Visual Studio Code (VS Code) plugins like **Cline**
- https://kiro.dev (a fork of VS Code)
- CLI tools like Amazon Kiro CLI for Mac and Linux) ← *my favorite for many tasks!*

–Note: There are many more! These are just some of my favorites for ham radio applications.

Summary

Get out of your lane! Explore new opportunities to build!

- Generative AI Large Language Models (LLMs) can understand and generate text to assist with many tasks.
- Model Context Protocol (MCP) connects AI models privately to your own data and tools.
- Prompt engineering is your main skill for getting high-quality results from AI. ***This is a key skill to learn!***
  Clear, detailed, and contextual prompts lead to better, more useful responses.
- Combining LLMs, MCP, and strong prompting turns generative AI from a curiosity into a *practical partner.*

→ **AI is like any other tool.** You have to learn to use it. Let's stop bashing AI and put it to work for us!

→ **YOU** can use generative AI to turn your ideas into real-world projects and share them with the world–*even if you have **zero** coding experience*.