Google Antigravity

AI model selection for RPG games and MQL4/MQL5

Reference guide • November 24, 2026

Choose models by visual intelligence, creative design, programming, mathematical reasoning and debugging. The recommendations below reproduce the earlier discussion; model names, availability and capabilities have not been independently verified for this document.

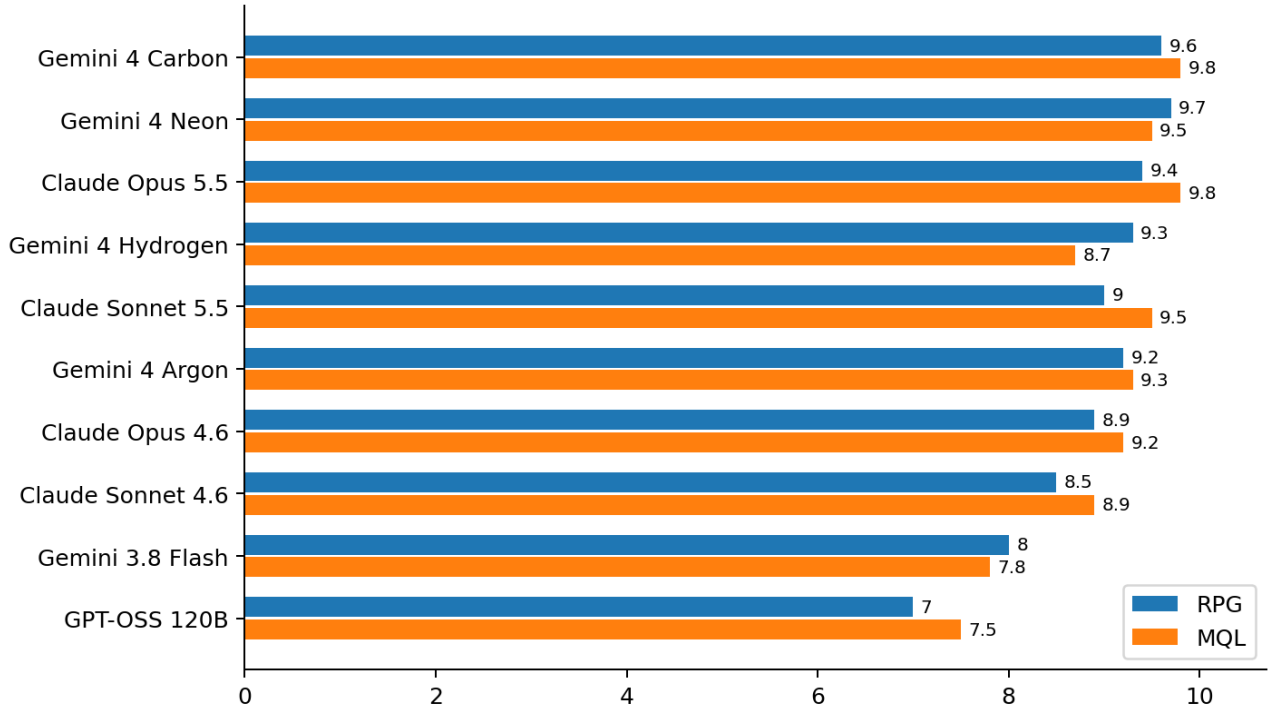
1. Model overview

Model	Suggested specialty
Gemini 4 Carbon	Deep reasoning and architecture
Gemini 4 Neon	Advanced coding and visual design
Claude Opus 5.5	Complex programming and analysis
Gemini 4 Hydrogen	Fast everyday execution
Claude Sonnet 5.5	Balanced coding and refactoring
Gemini 4 Argon	Advanced reasoning and debugging
Claude Opus 4.6	Complex programming
Claude Sonnet 4.6	General programming
Gemini 3.8 Flash	Simple fast tasks
GPT-OSS 120B	General reasoning
Orchestrator	Multi-model collaboration

The earlier answer mentioned a November 30 retirement date for several older models. Confirm current availability in the model picker and official documentation before relying on that date.

2. Estimated model ratings

Subjective suitability scores out of 10, not official benchmark results or tests on your code.



3. RPG development

Neon Extra: advanced graphics, room design and gameplay. Hydrogen Extra: routine room construction, GLB placement and decoration. Carbon High: architecture, pathfinding and performance. Opus Thinking: persistence, inventory and complex bugs.

Hydrogen Extra has already been useful for your Queen’s bedroom, dining room and library. Continue with it for routine rooms and try Neon for more elaborate scenes.

RPG task reference

Task	Suggested model
Recreate a room from a picture	Neon Extra
Queen's palace / 3D lighting	Neon Extra
GLB decoration / dungeon rooms	Hydrogen Extra
NPC AI / pathfinding	Carbon High
Combat system	Neon Extra
Inventory / JavaScript fixes	Sonnet 5.5
Permanent wardrobe / save-load	Opus 5.5
Three.js FPS optimization	Carbon High
Major upgrades	Orchestrator

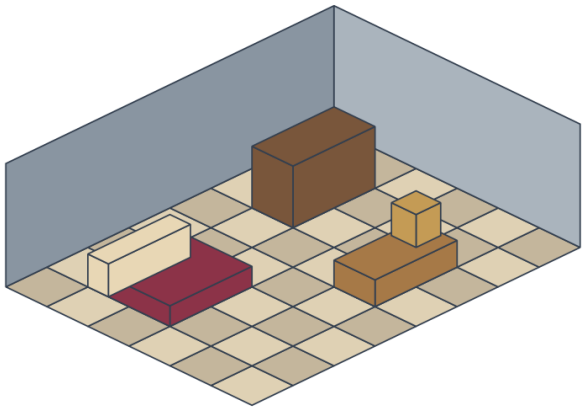


Illustration: a stylized isometric room layout for asset placement and level design (not an actual KayKit screenshot).

4. MQL4 and MQL5 programming

Opus 5.5 Thinking: complex indicators, EAs and non-repainting logic. Carbon High: mathematical algorithms, multi-timeframe logic and optimization. Sonnet 5.5 Thinking: everyday coding and compiler fixes.

Task	Suggested model
Standard MT5 indicators	Sonnet 5.5
Complex custom indicators / EAs	Opus 5.5
MQL4 to MQL5 / Pine conversion	Opus 5.5
Compilation errors	Sonnet 5.5
Repainting arrows	Opus 5.5
Multi-timeframe indicators	Carbon High
Fibonacci / Andrew's Pitchfork	Carbon High
Market Profile optimization	Carbon High
Camarilla Pivot	Sonnet 5.5
Backtesting logic	Carbon High
MT5 graphical dashboards	Neon Extra
Simple RSI, ATR, Stochastic	Hydrogen High
Reduce CPU usage	Carbon High

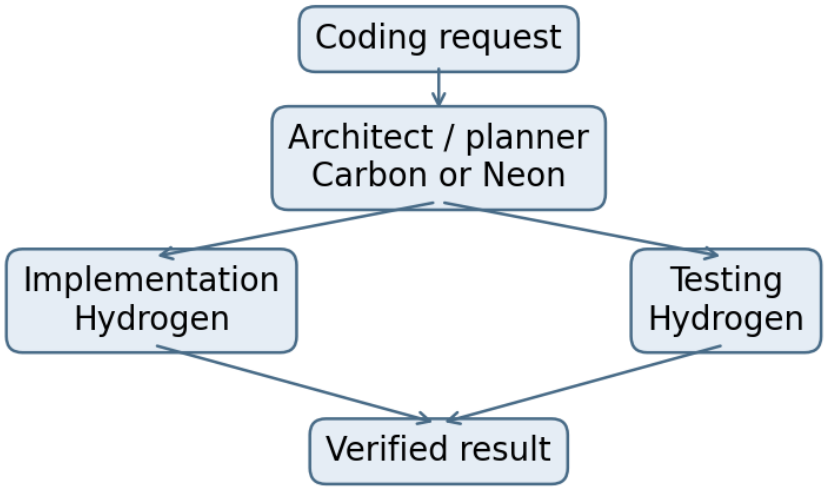
5. Thinking levels

Level	Use
Low	Minor changes and simple edits
Medium	Everyday programming
High	Complex programming
Extra	Difficult reasoning and debugging

Higher effort is not always necessary. Check which levels your selected model actually supports.

6. Orchestrator

Illustrative workflow for coordinating planning, implementation and testing. Actual model routing may differ.



7. Recommended setup

Work	Suggested model
Everyday RPG room design	Hydrogen Extra
Advanced graphics and gameplay	Neon Extra
Everyday MQL coding	Sonnet 5.5 Thinking
Complex Forex indicators and EAs	Opus 5.5 Thinking
Mathematics and optimization	Carbon High
Large software upgrades	Orchestrator

Suggested quota allocation: 70% Hydrogen, 20% Neon or Sonnet, 10% Carbon or Opus. This is a heuristic, not a measured quota estimate. Always compile in MetaEditor and test in Strategy Tester before live trading.

Official reference to check: https://antigravity.google/docs/models?app=antigravity