FlowGrid Design — User Guide

Version: 1.0 · **Date:** February 2026 · **Audience:** IT Architects, Technical Leads, Solution Designers

Table of Contents

1. Quickstart Guide

Get FlowGrid Design running locally in under 5 minutes.

Prerequisites

- **Docker Desktop** (v24+) with Docker Compose v2
- **Anthropic API Key** — a valid `ANTHROPIC_API_KEY` for Claude Sonnet (the AI engine behind the wizard)
- A modern browser (Chrome, Firefox, Edge)

Setup

```
# 1. Clone the repository
git clone https://github.com/your-org/flowgrid-platform.git
cd flowgrid-platform

# 2. Configure environment
cp infrastructure/.env.example infrastructure/.env
# Edit .env and set:
#   ANTHROPIC_API_KEY=sk-ant-...
#   JWT_SECRET=<any-random-string>

# 3. Start all services
cd infrastructure
docker compose up -d
```

Docker Compose starts the following services:

Service	Port	Purpose
nginx (Gateway)	8080	Entry point — serves Wizard UI, routes API calls
wizard-service	3005	AI-powered design wizard backend
design-module	3006	Visual network editor (three-panel UI)
agent-service	3001	Agent CRUD and registry
auth-service	3002	Authentication & JWT tokens
design-service	3003	Design storage and management
integration-service	3004	External system connectors
postgres	5432	Primary database
redis	6379	Caching and session state

First Run

1. Open **http://localhost:8080** in your browser
2. Register an account or log in
3. Navigate to the **Design Wizard** (`/wizard/`)
4. Follow the 8-step wizard to design your first agent swarm
5. After import, open the **Design Module** (`/design/`) to visualize and edit your network

> **Tip:** The wizard saves progress to localStorage automatically. If you close the browser, you can resume within 24 hours.

---

2. Workflow Overview

FlowGrid Design follows a **three-phase workflow**:

```
AI Wizard             Design Module          Export / Run
(8 steps)               (visual editor)          (future levels)
```

Phase 1: AI Wizard — Design Your Agent Swarm

The wizard guides you through **8 steps** of process-first agent design:

Step	Name	What Happens
1	**Process**	Select a Discovery Foundation and choose a process to agentize
2	**Sub-process**	Describe the specific workflow, expected outcomes, and constraints
3	**Agents**	AI identifies agents, applies the Abstraction Test, and optimizes the swarm
4	**Review**	Review and edit proposed agents — change names, patterns, autonomy levels
5	**Flows**	AI generates BPMN workflows: one orchestrator flow + per-agent internal flows
6	**Review Flows**	Inspect generated BPMN diagrams in an interactive viewer
7	**Config**	AI generates A2A-compliant configurations, skills, and integration specs
8	**Import**	Review summary and import the swarm into the Design Module

> **Key Insight:** The wizard uses a *process-first* approach. You start with a business process, and the AI derives the right agents — not the other way around.

Phase 2: Design Module — Visualize & Edit

After import, the Design Module provides a **three-panel editor**:

- **Left Sidebar** — Element list (agents, capabilities, data objects, processes) with search and filtering
- **Center Canvas** — Interactive network graph (vis-network) showing agents and their relationships
- **Right Panel** — Detail view for selected elements: properties, BPMN flows, relationships, skills

Phase 3: Export / Orchestrate (Future)

- **Level 2** (next): Export agent designs as deployable code (LangGraph, CrewAI, Azure Functions)
- **Level 3** (target): Live BPMN execution within FlowGrid itself

---

3. Objects & Concepts

3.1 Element Types

FlowGrid designs consist of four element types:

Type	Icon	Description	Example
Agent		An autonomous AI worker that can reason, decide, and act	Incident Triage Agent
Capability		A skill or ability an agent possesses	"Classify Incidents"
Data Object		An information store or data entity	Incident Record, CMDB
Process		A workflow or sequence of activities	Incident Lifecycle

3.2 Agentic Patterns

Each agent is assigned a **pattern** that defines its role in the swarm:

Pattern	Description	When to Use
Orchestrator	Coordinates other agents, manages the overall flow	Central coordination of multi-agent workflows
Specialist	Deep expertise in a narrow domain	Focused tasks requiring domain knowledge
Coordinator	Facilitates communication between agents	Cross-team or cross-domain coordination
Gateway	Entry/exit point, routes requests to the right agent	API facades, request routing
Monitor	Observes system state and triggers actions	SLA monitoring, anomaly detection
Executor	Performs concrete actions in external systems	ServiceNow ticket creation, deployments
Analyzer	Processes data and produces insights	Log analysis, trend detection
Aggregator	Combines outputs from multiple agents	Report generation, data consolidation
Router	Directs work to the appropriate agent based on rules	Intelligent request distribution

Note: During optimization, the AI also considers **Anthropic agentic patterns** (routing, planning, tool-use, orchestration, human-in-loop, RAG, reflection, guardrails) as implementation-level descriptors stored in agent config metadata.

3.3 Autonomy Levels

Every agent has an autonomy level that governs how much human oversight is required:

Level	Description	Example
Autonomous	Acts independently within defined boundaries	Auto-assigning low-priority tickets
Supervised	Acts but reports for review	Generating change plans for review
Human-in-Loop	Requires explicit human approval before acting	Approving P1 incident escalations

3.4 Human-in-the-Loop (HITL)

FlowGrid supports three HITL variants:

- **HITL** (Human-in-the-Loop) — Human must approve before the agent proceeds
- **HOTL** (Human-on-the-Loop) — Human is notified and can intervene, but the agent proceeds by default
- **HITM** (Human-in-the-Middle) — Human participates as a step within the workflow

The AI optimization step automatically identifies where HITL points should be added, especially around financial decisions, security changes, and compliance-sensitive operations.

3.5 BPMN Flows — Dual Level

FlowGrid generates BPMN at **two levels**:

Orchestrator BPMN (Inter-Agent)

- Defines **coordination between agents** using swim lanes
- Each lane represents an agent; service tasks map to agent invocations
- User tasks represent HITL approval points
- Gateways handle routing decisions

Agent Internal BPMN (Intra-Agent)

- Defines the **internal logic within a single agent**
- Steps → function calls, LLM invocations, tool usage
- Decision points → conditional gateways

- Error handling → boundary events

> **Why two levels?** Orchestration is coordination logic ("first A, then B, if X ask human") — perfect for a BPMN engine. Agent internals are cognitive work ("call LLM, parse response, reason") — better expressed as native code. This split enables independent evolution of coordination and agent logic.

3.6 Demoted Tools

Not everything needs to be an agent. The AI optimizer applies the **Abstraction Test**:

- **Does it need to reason?** → Keep as Agent
- **Just executes rules/lookups?** → Demote to Tool (a function on another agent)
- **Monitors over time?** → Move to async flow

Demoted tools appear as tools assigned to their parent agent, reducing swarm complexity and cost.

3.7 Risk Appetite

Each agent has a risk appetite setting (`low`, `medium`, `high`) that influences how much latitude it has for autonomous decision-making.

---

4. Step-by-Step Instructions

4.1 Running the AI Wizard

Step 1: Select a Process

1. Open the wizard at `http://localhost:8080/wizard/`
2. Select a **Discovery Foundation** from the dropdown — these are pre-analyzed capability maps of your organization
3. Browse the available processes within the foundation
4. Click a process card to select it — you'll see a preview with capability and process counts
5. Click **Continue →**

[Screenshot: Wizard Step 1 — Process Selection]

> **Tip:** If no foundations are available, you need to create one first via the Discovery module (`/discovery/`).

Step 2: Define Sub-Process

1. Enter a **Sub-process Name** (e.g., "Incident Triage & Assignment")

2. Describe **what should be automated** — be specific about inputs, steps, and expected behavior
3. Define the **Expected Outcome** — what does success look like?
4. Optionally add **Constraints** (e.g., "Must integrate with ServiceNow", "max 3 agents", "human approval for P1")
5. Click **Generate Agents**

[Screenshot: Wizard Step 2 — Sub-Process Definition]

> **Tip:** Click **Surprise Me** to auto-fill with a sample sub-process — useful for exploring the platform.

Step 3: AI Agent Identification The AI performs a multi-step analysis:

1. **Propose Agents** — Identifies candidate agents from your process description
2. **Optimize** — Applies the Abstraction Test: merges redundant agents, demotes simple ones to tools, identifies missing HITL points
3. **Assign Patterns** — Selects the best agentic pattern for each agent

A progress bar shows the current phase. This typically takes 15–30 seconds.

[Screenshot: Wizard Step 3 — AI Identifying Agents]

Once complete, you'll see agent cards showing: - Agent name and pattern badge - Purpose description - Key responsibilities - Orchestrator agents are highlighted in red

Step 4: Review & Edit Agents This is your chance to fine-tune before flow generation:

- **Edit** any agent's name, purpose, pattern, or autonomy level
- **Remove** agents you don't need
- **Add** new agents with the button
- Adjust **responsibilities** and **capabilities**

[Screenshot: Wizard Step 4 — Editable Agent Cards]

Click **Approve & Generate Agent Flows** when satisfied.

Step 5: AI BPMN Generation The AI generates BPMN flows:

1. **Orchestrator BPMN** — The coordination flow with swim lanes for each agent
2. **Per-agent internal BPMN** — Individual workflow for each agent's internal logic

Progress updates show which agent flow is being generated. This step takes 30–60 seconds depending on swarm size.

[Screenshot: Wizard Step 5 — Generating Flows]

Step 6: Review Agent Flows Interactive BPMN viewer with tabbed navigation:

- **Orchestrator** tab — Shows the full coordination flow with swim lanes
- **Individual agent** tabs — Each agent's internal workflow
- Use mouse wheel to zoom, drag to pan
- BPMN elements are rendered with the bpmn-js viewer

[Screenshot: Wizard Step 6 — BPMN Flow Review]

> **Note:** If a flow looks truncated or incorrect, go back to Step 5 and regenerate. AI output can vary between runs.

Step 7: Agent Configuration The AI generates A2A-compliant configurations:

- **Skills** — Each agent's capabilities with input/output schemas and examples
- **Tools** — Functions available to each agent (including demoted agents)
- **Interactions** — Relationships, message types, and integration points

Review the config cards for each agent.

[Screenshot: Wizard Step 7 — Agent Configurations]

Step 8: Import & Create Swarm

1. Review the final summary: agent count, flow count, skill count
2. Verify the agent list
3. Click **Import & Create Swarm**
4. On success, click **Open Design Module →** to view your swarm

[Screenshot: Wizard Step 8 — Import Summary]

4.2 Using the Design Module

The Design Module (`/design/`) is a three-panel visual editor.

Left Sidebar — Element List

- Lists all elements (agents, capabilities, data objects, processes)
- **Search** by name to filter
- Click any element to select it on the canvas and open its detail panel
- Element type icons help distinguish agents from capabilities

Center Canvas — Network Graph

- **Interactive vis-network graph** showing agents as nodes and relationships as edges

- **Orchestrator agents** are displayed larger and in red
- **Zoom** with mouse wheel, **pan** by dragging the canvas
- **Click** a node to select and view its details
- **Drag** nodes to rearrange the layout
- Relationship lines show message types and interaction patterns

Right Panel — Detail View When an element is selected, the right panel shows:

- **Properties** — Name, description, pattern, autonomy level, risk appetite
- **BPMN Flow** — Embedded bpmn-js viewer showing the agent's workflow (with full modeler capabilities including palette and editing tools)
- **Relationships** — Connected agents, message schemas, async/sync indicators
- **Skills** — A2A skill definitions with input/output schemas
- **Tools** — Available tools and integrations

Editing Agents In the Design Module you can:

- View and inspect all agent properties
- Browse BPMN flows for each agent
- See the full relationship network
- The BPMN modeler includes a palette for adding elements

Viewing BPMN Flows

- Select an agent → the BPMN viewer loads its internal flow
- Switch between orchestrator and agent-internal flows via tabs
- The viewer supports zoom, pan, and element selection

---

5. Current Limitations & Workarounds

FlowGrid Design is a **Level 1 platform** — a design tool. Here's what to expect:

Limitation	Impact	Workaround
AI output inconsistency	The same input may produce different agent proposals or BPMN flows across runs	Regenerate (go back a step and re-run), or manually edit agents in Step 4
Orchestrator BPMN swim lane rendering	Large flows may truncate or render swim lanes incorrectly	Regenerate the BPMN; review in the Design Module's bpmn-js modeler

Limitation	Impact	Workaround
No runtime execution	Agents cannot actually execute — this is design-only (Level 1)	Use exports (Level 2, coming soon) to generate deployable code
No Docker sandbox	No containerized agent execution environment	Planned for Level 3
Single AI model	All prompts use Claude Sonnet — no per-prompt model selection	Planned: per-agent model configuration
No import/export of designs	Cannot save designs as files or import from other tools	Use the database-backed design store; file export planned
BPMN editing	The Design Module includes bpmn-js with modeler palette, but changes may not persist back to the agent model	Edit agent descriptions in the wizard and regenerate flows
Foundation dependency	The wizard requires a Discovery Foundation; you cannot start from scratch without one	Create a foundation via the Discovery module first

> **Tip:** The AI is good at getting you 80% of the way. Expect to manually review and adjust agent designs, especially for complex processes with many agents.

---

6. Platform Roadmap & Forward-Looking Statement

Level 1 (Current) — Design Tool

AI-powered agent network design via 8-step wizard Visual Design Module with three-panel editor Dual BPMN generation (orchestrator + agent-internal) A2A-compliant agent configurations Optimization engine (Abstraction Test, HITL detection, tool demotion)

Level 2 (Next) — Design + Export

- **Code Generation Engine** — Export agent designs as deployable code
- **Target frameworks:** LangGraph (primary), CrewAI, Azure Functions
- **Dual export modes:**
  - *Agent-only mode* — generates agent internal code from BPMN (for users planning to use FlowGrid's own orchestrator at Level 3)
  - *Standalone mode* — generates both agent code AND orchestrator code (for self-contained deployments)
- Template-driven, deterministic code generation (no AI in the export path)
- Static validation of generated code before packaging

Level 3 (Target) — Design + Orchestrate

- **Live BPMN execution** via `bpmn-engine` (Node.js)
- FlowGrid becomes the **control plane** for multi-agent systems:
  - Flow orchestration (BPMN runtime)
  - State management (shared state store)
  - Human-in-the-loop approval queues
  - Event routing (agent-to-agent communication)
  - Task dispatch with retry and dead-letter queues
  - Execution traces and observability dashboards
- Agent execution delegated externally (LLM inference via APIs)
- Hot-reloadable orchestration flows

Future Considerations

- **Hybrid BPMN + dynamic agent orchestration** — Combine structured BPMN flows with dynamic LLM-based routing for complex scenarios
- **Per-agent model configuration** — Choose different LLMs for different agents (Claude, GPT-4, Gemini, local models)
- **CrewAI integration research** — Evaluate CrewAI as an alternative orchestration backend
- **Multi-tenant execution** — Isolated agent runtimes per tenant with Azure Service Bus bridging
- **Agent Registry** — A2A-compliant agent discovery and marketplace

---

Forward-Looking Statement

> *The features and timelines described in the roadmap section represent our current plans and intentions. They are subject to change based on technical feasibility, user feedback, and business priorities. No commitment is made to deliver any specific feature by any specific date. Level 1 (Design Tool) is the current generally available capability.*

Appendix: Architecture Overview

```
                    nginx (Gateway :8080)
          Wizard UI · Design Module · API Routing

  wizard     design     agent      auth      integration
  service    module     service    service     service
  :3005      :3006      :3001      :3002       :3004

               PostgreSQL :5432  ·  Redis :6379
```

All services communicate over a Docker bridge network (`flowgrid-network`). The nginx gateway handles routing, static file serving, and SSL termination.

© 2026 FlowGrid Platform. Built with for IT architects who believe in designing agent systems properly.