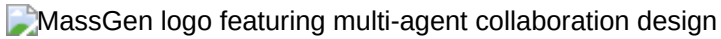
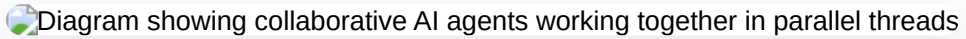




Scaling AI Through Multi-Agent Collaboration

🎓 **M2L Summer School**
📍 Split • September 11, 2025

🌐 massgen.ai | GitHub

The Single-Agent Limitation

- **Siloed Thinking:** Single models miss diverse perspectives
- **Limited Context:** No peer review or validation
- **Sequential Processing:** Linear, not parallel exploration
- **Fixed Approach:** Limited mid-task adaptation to new information

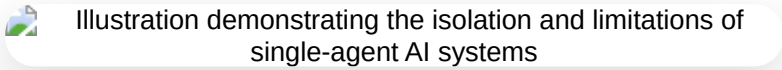


Illustration demonstrating the isolation and limitations of single-agent AI systems

From Isolation to Collaboration

🤝

The Promise of Multi-Agent Collaboration

- → **Study Group Dynamics:** Like humans collaborating on complex problems
- → **Cross-Model Synergy:** Leverage unique strengths of Claude, Gemini, GPT, Grok
- → **Parallel Processing:** Multiple perspectives tackle same task simultaneously
- → **Real-time Intelligence Sharing:** Agents learn and adapt from each other

Visual representation of collaborative AI reasoning and cognitive processes

The Promise of Collaborative Reasoning

📖 root.massgen.ai - "Myth of Reasoning"

🏗️ Built on AG2's foundational multi-agent research and community

AG2: The Foundation for Multi-Agent Research

🤝 Community-Driven Innovation

MassGen evolved from AG2's pioneering work in multi-agent conversations
and the vibrant research community it fostered

Proven Performance Gains - Grok Heavy Evidence

Grok-4 Standard

1

Single Agent Processing

38.6%

Last Human Exam Score

$30/month

Grok-4 Heavy

A1 A2 A3

Multi-Agent Collaboration

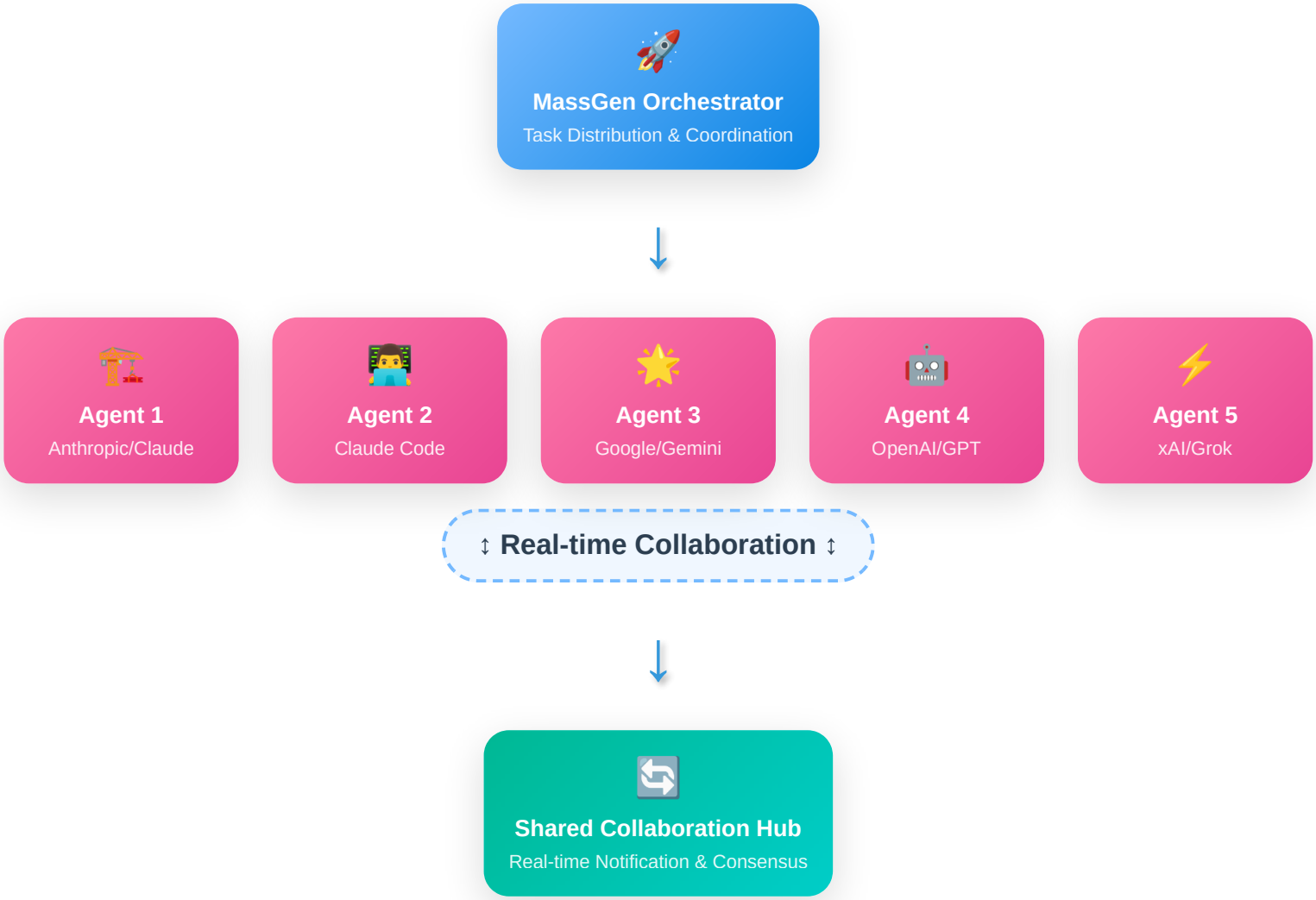
44.4%

Last Human Exam Score

$300/month

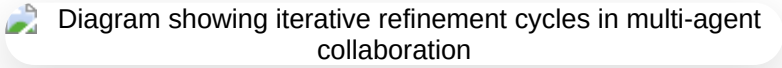
+15% Performance Boost

Multi-agent "study group" approach outperforms single agent


MassGen Orchestrator
Task Distribution & Coordination
Agent 1
Anthropic/Claude
Agent 2
Claude Code
Agent 3
Google/Gemini
Agent 4
OpenAI/GPT
Agent 5
xAI/Grok
↕ Real-time Collaboration ↕
Shared Collaboration Hub
Real-time Notification & Consensus

Key Features & Capabilities

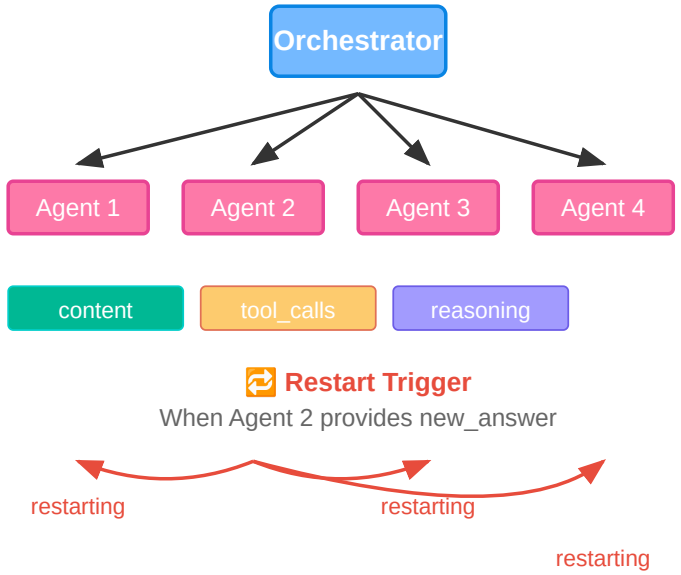
- 🤝 **Cross-Model Synergy:** Harness strengths from diverse models
- ⚡ **Parallel Processing:** Multiple agents tackle problems simultaneously
- 🔄 **Iterative Refinement:** Non-linear reasoning through cycles of improvement
- 👥 **Intelligence Sharing:** Agents share working summaries, tool results, and insights in real-time
- 🎯 **Consensus Building:** Natural convergence through collaboration



Iterative Refinement: The Reality of Reasoning

Tech Deep Dive: Async Streaming & Dynamic Scheduling

- → 🔄 **AsyncGenerator Pattern:** Real-time streaming from 5+ agents simultaneously
- → ⚡ **Dynamic Task Management:** Agents start/stop based on voting status
- → 🔁 **Graceful Restart & Wrap-up:** Dynamic wrapping-up as part of scheduling



Key Innovation: Dynamic coordination without deadlocks

Tech Deep Dive: Backend Abstraction Challenges

- 🎭 **Unified Interface:** Standardized ChatAgent protocol for 8+ different backends
- 🛠️ **Tool Integration:** Web search, code execution, MCP
- ⚙️ **StreamChunk Normalization:** Convert diverse response formats to common protocol
- 🔀 **Backend-Specific Workarounds:** Each provider has unique limitations

Backend Challenges:

Claude Code CLI Context sharing across agents	**Gemini API** Can't mix builtin + custom tools
GPT-5 API change (reasoning, streaming etc.)	**Most Backends** Unable to autonomously collaborate

🎯 **Our Solution:**
Binary Decision Framework & Advanced Workspace Sharing

Result: Unified interface with backend-specific optimizations

Tech Deep Dive: Binary Decision Framework Solution

- ⚖️ **Binary Choice:** Each agent must choose: vote OR new_answer
- 💥 **Vote Invalidation:** Any new_answer invalidates ALL existing votes
- 🔄 **Reset & Restart:** All agents restart with updated answer context
- 🎭 **Anonymous Voting:** Agents see "agent1", "agent2" etc.

Key Innovation: Dynamic equilibrium through vote invalidation

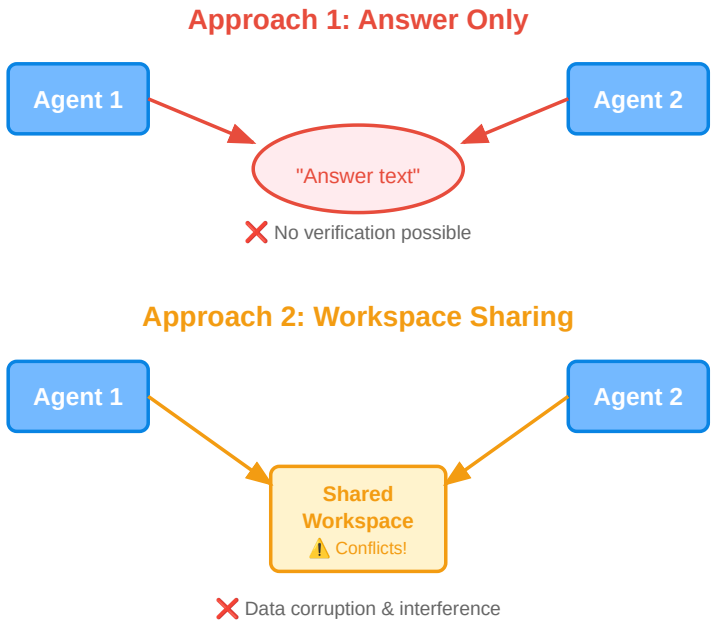
The Context Sharing Challenge

❌ Naive Approach 1: Share Answers Only

- → Agents only see final text answers
- → Can't verify methodology or data
- → Unable to test or build upon work
- → Lost intermediate context

❌ Naive Approach 2: Share Workspace Paths

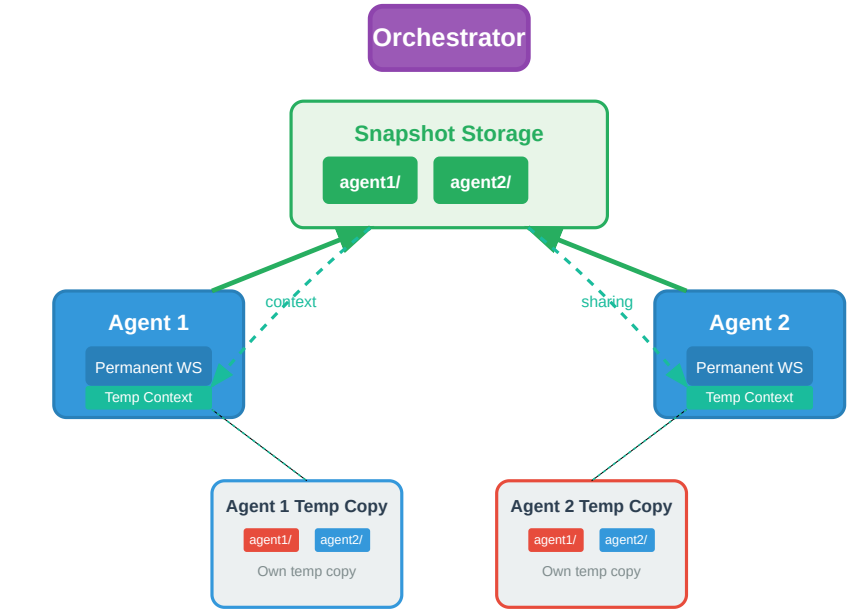
- → Agents interfere with each other's work
- → Data corruption from simultaneous edits
- → Loss of original work context
- → Workspace pollution and conflicts



The Challenge: How to share context without interference?

Our Context Sharing Solution

- → 📸 **Workspace Snapshots:** Orchestrator captures agent workspaces after each round
- → 📁 **Temporary Directories:** Each agent gets a clean temp workspace with all snapshots
- → 🎭 **Anonymous Mapping:** agent1/, agent2/ folders preserve anonymity
- → 🔒 **Clean Separation:** Read from temp dir, write to permanent workspace
- → 🔄 **Context Preservation:** Snapshots linked to coordination rounds



✅ Safe Context Sharing + ✅ No Interference + ✅ Full Verification

Context Sharing in Action

🔬 Round 1: Agent 1 (Data Scientist)

- Creates `analysis.py` and `results.csv`
- Saves to permanent workspace
- 📸 **Snapshot captured**

🧪 Round 2: Agent 2 (Code Reviewer)

- Sees `agent1/analysis.py` in temp workspace
- **Reads & tests** the analysis code
- Modifications in temp dir **don't affect** Agent 1
- Creates `improved_analysis.py` in own workspace

🎯 Final Presentation

- Winning agent has **full context**
- Can reference both agents' work
- Snapshots ensure **correct version** access

🗂️ Workspace Structure

👨‍💼 Agent 1 Permanent Workspace

```
📄 analysis.py
📊 results.csv
📝 methodology.md
```

👁️ Agent 2 Temp Workspace (Read-Only Context)

```
📁 agent1/
   📄 analysis.py ✅ readable & testable
   📊 results.csv
   📝 methodology.md
```

👩‍💻 Agent 2 Permanent Workspace

```
📄 improved_analysis.py
📋 code_review.md
🧪 test_results.json
```

🔑 Key Benefits Illustrated:

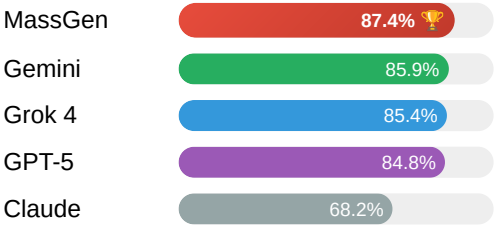
✅ Agent 2 can READ & execute Agent 1's work
✅ Temp modifications don't corrupt original
✅ Each agent maintains workspace integrity
✅ Final answer has complete context

Benchmarking: Preliminary Results

Scientific evaluation across graduate-level reasoning, instruction-following, and narrative tasks

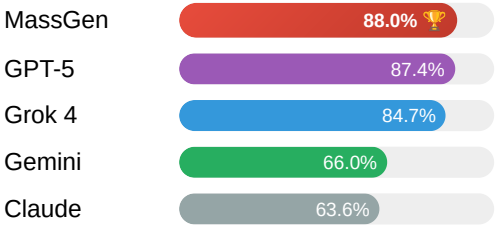
🧪 GPQA-Diamond

Graduate Physics/Chemistry



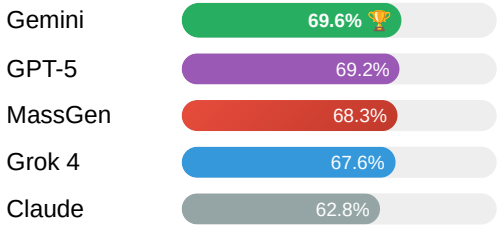
📋 IFEval

Instruction Following



📖 MuSR

Narrative Reasoning



🏆 Overall Champion

MassGen: 81.2%

Wins 2/3 benchmarks • Statistically significant

✅ Key Results:

- **Highest on 2/3** benchmarks
- **Best overall** average
- **Consistent** performance

📈 Statistical:

- vs Claude: **p = 1.4e-07** ⭐⭐⭐
- vs Gemini: **p = 1.1e-28** ⭐⭐⭐
- Not due to chance

🔬 Research Gap:

- **Oracle:** 95.5% (GPQA)
- **Actual:** 87.4%
- **Potential:** 8.1 points

Case Study: Success Through Peer Correction

Graduate-level physics question from GPQA-Diamond benchmark

🌌 The Problem

A quasar shows a peak at 790 nm wavelength. Given Lambda-CDM cosmological parameters (H_0 = 70 km/s/Mpc, Ω_m = 0.3, $\Omega\Lambda$ = 0.7), what is the comoving distance?

Options: A) 8 Gpc B) 7 Gpc C) 6 Gpc D) 9 Gpc

🤖 Round 1: Initial Answers

Claude: "I calculate ~6 Gpc → Answer C"
GPT-5: "I get ~8.95 Gpc → Answer D"
Gemini: "~6.1 Gpc → Answer C"

🔄 Self-Correction Process

Claude observes: *"There is significant discrepancy in calculations: Agent1 gets ~6.1 Gpc, Agent2 gets ~8.95 Gpc. Let me re-examine..."*

🎯 Final Result

✅

Correct Answer: A (8 Gpc)

Orchestration succeeded where individual agents initially failed

✨ Breakthrough Moment

Claude revises: *"Standard cosmological calculators yield 8000-8500 Mpc for z=5.5. This equals 8.0-8.5 Gpc, closest to option A."*

Result: *3/4 agents converge on correct answer*

💡 **Success Mechanism:**
Peer observation → Discrepancy detection → Self-correction → Consensus