Automaton Integration Strategy

Document: 242-PP-RMAP | **Category**: PP (Project Planning) | **Type**: RMAP (Roadmap) **Date**: 2026-02-18 | **Author**: Intent Solutions **Status**: Strategic Recommendation

Purpose

Evaluate three integration strategies for Conway's Automaton within the Intent Solutions ecosystem. Recommend an approach that maximizes synergy while managing risk.

Background

Automaton is a self-sovereign AI agent framework created by Conway Research (forked, not original work). It earns its own existence through work, can replicate, and self-modify. It uses ERC-8004 identity, USDC payments, and runs on Conway Cloud infrastructure.

Option A: Standalone Sovereign Agent (Minimal Integration)

Run Automaton as-is on Conway Cloud. Let it earn its own way. Monitor from a distance.

Architecture

```
Conway Cloud
    Automaton (standalone)
          SOUL.md (constitution)
          Social Inbox
          USDC Wallet
          Self-contained tools
```

Assessment

Factor	Rating	Notes
Engineering effort	None	Zero integration work
Risk	Low	Isolated from existing systems
Learning value	Medium	Observe autonomous agent behavior
Revenue potential	Low	No leverage of existing capabilities
Strategic value	Low	No synergy with ecosystem

Verdict

Safe but limited. Good for initial observation phase only.

Option B: Automaton as Broker Front-End to GWI (Medium Integration)

Wire GWI as tools in Automaton. Broker receives work requests via social inbox, delegates to GWI agents, collects USDC.

Architecture

```
Clients (agents, humans)


    AUTOMATON    ← social inbox, x402 payments
     (Broker)



      GWI        ← PR triage, review, resolve
    MCP/API
```

Assessment

Factor	Rating	Notes
Engineering effort	Low	5 tool wrappers + `--json` flag work
Risk	Medium	Single integration point
Revenue potential	Medium	Monetizes GWI through autonomous agent
Strategic value	Medium	Tests agent economics model

Verdict

Viable first step. Tests economics without full stack commitment. But leaves money on the table by not using IRSB/Moat.

Option C: Automaton + IRSB + Moat Stack (Deep Integration) — RECOMMENDED

The full play: Automaton runs on Conway Cloud but its wallet delegates to IRSB's WalletDelegate for on-chain guardrails. Moat enforces policy on every capability call. Bob's Brain provides risk tier governance. GWI/Products are capabilities the broker sells.

Architecture

```
                    CLIENTS (agents, humans)


                      AUTOMATON    ← social inbox, x402 payments
                      (Broker)     ← SOUL.md, survival tiers



          IRSB        MOAT        BOB'S
```

```
on-chain    policy      BRAIN
guardrails  receipts   risk tiers
```

```
  GWI      PRODUCTS   PERCEPTION
PR auto    crypto     news intel
MCP srv    agents     226 feeds
```

Assessment

Factor	Rating	Notes
Engineering effort	High	Multi-system integration across 5+ projects
Risk	Medium	Staged rollout mitigates blast radius
Revenue potential	High	Full capability set monetized through autonomous agent
Strategic value	Very High	Creates the Intent Agent Network reference implementation

Why This is the Recommended Approach

1. **Defense in Depth**: Three independent safety layers (on-chain, policy, governance) ensure no single compromise defeats all protections
2. **Economic Alignment**: IRSB creates real economic friction for each agent action, preventing unconstrained spend
3. **Audit Trail**: Moat receipts + IRSB on-chain receipts create dual audit trails (off-chain + on-chain)
4. **Graduated Autonomy**: Bob's Brain risk tiers allow the agent to operate freely for low-risk tasks while requiring human approval for high-risk operations
5. **Reference Implementation**: This becomes the blueprint for the Intent Agent Network — every future agent follows the same pattern

Verdict

Higher upfront investment, but creates lasting infrastructure that benefits all future agents. This is the strategic play.

Implementation Roadmap

Phase 1: Wire Automaton to GWI (1-2 days)

- Add 5 GWI tool wrappers to `agent/tools.ts`
- Create GWI integration SKILL.md
- Test with a real PR triage request
- **Files**: `99-forked/automaton/src/agent/tools.ts`

Phase 2: IRSB Wallet Delegation (1 week)

- On Automaton boot, delegate wallet to IRSB WalletDelegate
- Configure SpendLimitEnforcer (daily cap for compute costs)
- Configure AllowedTargetsEnforcer (Conway Cloud + GitHub only)
- **Files**: `99-forked/automaton/src/identity/wallet.ts`, `irsb-monorepo/protocol/`

Phase 3: Moat Policy Layer (1-2 weeks)

- Register Automaton's capabilities in Moat Control Plane
- Route all tool calls through Moat Gateway
- Enable receipt generation for audit trail
- **Files**: `moat/`, `99-forked/automaton/src/agent/tools.ts`

Phase 4: Bob's Brain Governance (2-3 weeks)

Define risk tier mapping for Automaton operations:

Tier	Operations	Approval
R0	Read-only queries, triage	Autonomous
R1	Code review, analysis	Autonomous
R2	PR creation, branch push	Autonomous with logging
R3	Merge, deploy	Requires human approval
R4	Financial transactions	Requires human + IRSB receipt

Phase 5 (Optional): Bob Refactor to TypeScript

- Port Bob's Brain orchestrator from Python/ADK to TypeScript
- Align with GWI/Automaton stack
- Maintain Vertex AI Agent Engine integration via REST API

Decision Matrix

Criterion	Weight	Option A	Option B	Option C
Strategic value	30%	1	5	9
Revenue potential	25%	2	6	8
Risk management	20%	8	5	7
Engineering effort	15%	10	7	4

Criterion	Weight	Option A	Option B	Option C
Learning value	10%	4	6	9
Weighted Score		**3.9**	**5.6**	**7.6**

Recommendation: Option C (Deep Integration) with phased rollout starting from Phase 1. Each phase is independently valuable and can be paused if priorities shift.

Document follows 6767 Filing Standard v4.2