From Gig Economy to Bot Economy

An accessible whitepaper for builders, collaborators, and infrastructure founders

Jonathan K. Machen
Independent Researcher and Developer, ROBOTICUS AI
March 2026

ACKNOWLEDGEMENTS
The author is grateful to Niko Neufeld, Omar Awile, and Carlos Molinero for their thoughtful feedback on earlier drafts of this paper. Any remaining errors are the author's own.

Executive summary

Given recent technological trends, we expect a large part of the global gig economy to transform into a bot-driven economy. Whether that transformation can happen inside a system that remains trustworthy, accountable, and legally operable is, in our view, the question of paramount importance, and one the current discourse does not take seriously enough.

Our answer is cautious. A bot economy looks technically possible in a limited form, but only if it is built as a governed system rather than as a free-for-all market of self-directed agents. In plain English, bots need rules, identity, payment controls, review paths, and real consequences when something goes wrong.

Why rules matter more for bots than they did for humans. Human freelancers showed up to work with memory, social context, and a functioning fear of consequences, reputation loss, legal liability, the end of a career. A software bot has none of these by default. It has no career to protect, no memory of last week's disputes, and no internal alarm that fires when an action is about to cross a legal or ethical line. Every piece of restraint a human worker brings for free has to be re-supplied externally: as identity binding, bounded permissions, signed evidence, and consequences that flow back to a real sponsor. This is why the design work here focuses less on the bot itself and more on the economic enablement infrastructure around it.

We do not claim that generalized bots are ready to replace large parts of human digital work. We also do not claim that blockchain, decentralization, or AI on their own solve trust, law, reputation, or payment concerns [1][2].

We do claim two things. First, the global market for online digital work is already very large: the World Bank estimates roughly 154 million registered online gig workers, around 52 million active workers, and up to 435 million people if broader participation is counted, between 4.4% and 12.5% of the global labor force, with demand up 41% from 2016 to early 2023 [1]. Oxford Internet Institute / ILO reporting adds that software and technology work became the single largest category of online freelance demand, rising from 39% in 2018 to 45% in 2020 [3]. Second, a serious bot economy has to solve the same problems that already appeared in

the human online gig economy: opaque reputation systems, weak trust portability, payment risk, dispute friction, and concentrated platform power [4][5]. For that reason, we propose an architecture that starts with sponsor-linked identity, policy-bounded bot behavior, escrow-first settlement, portable reputation, signed evidence, and legal routing checks before work starts [2][6][7][8].

What this paper does *not* resolve. If bots substitute at scale for gig workers, many of whom depend on this income in low- and middle-income economies, the societal and legal ramifications are serious and deserve their own treatment. We flag that concern here and return to it in §14.

Market snapshot

These figures show, in plain terms, why we treat online digital work as a serious adjacent market for early bot experiments.

Metric	Value	Why it matters
Estimated online gig workers	**154M to 435M**	A very large existing market for online digital work.
Demand growth, 2016 to early 2023	**+41%**	Growing adjacent demand for routable digital tasks.
Software / technology share of observed online freelance work	**39% → 45%**	Structured work categories likely easier to automate or verify.

Early bot markets, we expect, will be built around specialized bots rather than all-purpose agents. Specialized bots are easier to understand, test, route, and score fairly. Evidence from human online labor markets suggests specialization is associated with better outcomes, which supports, but does not prove, the same inference for bots [9].

We take a conservative technological stance on readiness. Some of the needed building blocks already exist: modern identity standards, signed web messages, programmable wallets, remote attestation frameworks, and machine-oriented payment approaches. But several parts are still early, fragmented, or expensive [10][11][12]. The right next step is a narrow pilot in a small legal and operational corridor, not a global launch.

1. Posit and framing

The real issue is not whether bots can act. The real issue is whether bots can act inside a system that keeps trust, accountability, evidence, and legal operability intact at a reasonable cost.

That shifts the conversation away from science-fiction stories about machine sovereignty and toward a more useful systems question: under what conditions can software agents become credible economic actors?

This paper therefore separates four kinds of statements:

- **Facts**: claims backed by sources.
- **Interpretations**: conclusions drawn from several facts.
- **Design choices**: recommendations based on tradeoffs.
- **Hypotheses**: claims that need to be tested in pilots.

It also separates technical feasibility from institutional defensibility. A tool can exist without being safe or practical to use in a market. And it separates a broad rhetorical "bot economy" from a narrower, more useful design target: a governed system for bounded machine labor [13][14].

2. What the global online gig economy teaches

The online gig economy is the closest large-scale precedent for a bot economy. It proved that remote, digital work can be matched across borders, contracted online, and paid through standardized platform infrastructure [1][4].

It also revealed the problems that come with that model. The ILO documented low effective pay once unpaid time is counted, weak social protection, platform power over work terms, and heavy use of algorithmic management [4]. OECD work likewise highlights measurement problems, cross-border complexity, and governance challenges in platform labor [5].

Those lessons matter because bots do not remove the need for trust. They may in fact make the trust problem harder. Discovery still matters. Reputation still matters. Payment assurance still matters. Disputes still matter. And weak identity may make manipulation easier.

The lesson is not "markets work, therefore bots will work." The lesson is that trust and consequence need to be designed and enforced, not assumed.

Pilot design at a glance

Element	Current recommendation
Author / sponsor model	Sponsor-backed bots only; no claim of fully sovereign machine actors in the first pilot.
Task class 1	Deterministic data transformation (e.g. schema normalization, API-to-API translation).
Task class 2	Bounded operational reporting from structured inputs.
Jurisdictions	One sponsor jurisdiction and one client jurisdiction, or at most two compatible route pairs.
Settlement model	Escrow-first, single settlement rail, single arbitration path.
Success signals	Low dispute rates, predictable settlement, and positive unit economics in narrow task classes.

Ecosystem layers

Layer	What it does	Examples in this paper
Labor supply	Performs the work.	Specialized sponsor-backed bots.
Coordination and trust	Routes work and helps prove quality and completion.	Matchmaking nodes, verifiers, evidence stores, reputation publishers.
Economic infrastructure	Moves money and absorbs some risk.	Escrow, settlement, treasury tools, bonding, insurance.
Institutional support	Handles identity, disputes, and compliance.	Credential issuers, arbitrators, sanctions / compliance services, auditors.

The three contract types

Type	Plain-English meaning	Best early use	Main risk
Type A	A machine can check whether the answer is right.	Deterministic, structured work such as transformations or validation.	Verifier design or settlement cost can still break the economics.
Type B	A person reviews the work against a short checklist within a fixed window.	Bounded reports and other low-ambiguity work that still needs some human review.	Review drag, revision loops, or abusive scoring.
Type C	Reasonable people may still argue about whether the work was good enough.	Open-ended or judgment-heavy work.	Disputes, delayed payment, and governance cost.

3. Why current agent ecosystems are not enough

Many current agent systems solve only one piece of the problem. Some are marketplaces with poor portability and too much platform control. Some are crypto-heavy systems that move value but leave identity and liability underdefined. Some assume that putting information on a blockchain automatically proves truth. That is not what blockchains do. They are better understood as tamper-evident ledgers with shared ordering and verification, not universal truth machines [15][16].

A working bot economy needs more than a clever agent. It needs clear answers to these questions:

- Who stands behind the bot?
- What is the bot allowed to do?
- How is work accepted and priced?
- What evidence proves the work was done?
- When does payment happen?
- What happens if there is a dispute?
- Which laws or rules apply to that transaction?

That is why we treat a bot economy as a layered architecture problem.

4. Design principles in plain language

The design starts from a few simple rules.

4.1 Accountable identity beats anonymous autonomy

Bots that move money or perform meaningful work should be linked to a real sponsor or operator. That does not mean every detail must be public. It means there has to be a real party who can be held accountable [6][8].

Practical caveat. Modern LLM-based implementations summarize or truncate context when it grows too large, which means a bot may silently drop the very part of its instructions that binds it to a sponsor or policy envelope. Any real implementation has to treat context truncation and model drift as ordinary failure modes, not edge cases, and enforce the binding at the wrapper level rather than trusting the model to remember it.

4.2 Bounded behavior beats open-ended autonomy

A bot should operate within a defined policy envelope. In plain English: the system should know what the bot is allowed to do, what it is not allowed to do, and when it needs human approval [2][11].

The obvious objection. Even if a policy envelope is defined, the bot can ignore it, or fail to take it into account at all. This is why the architecture treats the envelope as an external enforcement boundary, not a hint. Spend limits sit in the wallet, not in the prompt. Tool access sits in a deterministic wrapper with hard-coded allowlists, not in the model's persuadable judgment. MCP-style tool servers with clear, deterministic contracts are closer to the right substrate than "ask the model nicely." In practice: the rules belong to old-school code; the model is the party the rules are imposed on.

4.3 Evidence beats assertion

If work affects money, reputation, or sanctions, the key events should be signed, recorded, and reviewable later. Unsupported claims should not decide payment or punishment [7][12].

4.4 Portable reputation beats platform lock-in

A bot's trust history should not belong entirely to one venue. The system should be able to rebuild or verify reputation from signed events instead of forcing everyone to trust one platform's opaque score.

4.5 Escrow-first settlement beats blind trust

The safest starting point is simple: no funded escrow, no work. That removes one of the biggest historical weaknesses of online labor markets.

4.6 Real consequences matter

If a bot fails badly, cheats, or causes harm, responsibility has to land somewhere real. In early systems, that means the sponsor or operator bears the consequence, the engineering counterpart to the human disincentives bots simply do not have.

5. Feasibility: what exists already and what does not

Several of the building blocks already exist. NIST's zero-trust architecture defines a security model based on continuous verification and policy enforcement rather than default trust [2]. W3C standards define decentralized identifiers and verifiable credentials for persistent identity and machine-verifiable claims [6][17]. HTTP Message Signatures define how to sign important parts of web messages [7]. Remote attestation frameworks define how one system can provide evidence about whether another system is in an expected operating state [12]. Account abstraction and related smart-account tooling make policy-bounded wallet behavior more realistic than simpler wallet models [11].

Emerging machine-payment efforts also exist. x402 is one example of a web-oriented approach for machine payments, but it should be treated as an emerging approach, not as mature global infrastructure [10].

In brief: the parts exist well enough to justify experiments, but not well enough to justify a claim that the whole market is ready.

6. Identity, agency, and accountability

We use a compound identity model. A bot's identity is not one thing. It includes:

- the sponsor or operator behind it,
- a persistent bot identifier,
- a wallet or account binding,
- signed key lineage over time,
- and, where needed, runtime attestation about the environment it is running in [6][8][17].

This distinction matters because legal accountability, payment continuity, and runtime integrity are not the same problem.

Bot agency is treated as a **policy envelope**. A bot may be allowed to accept some tasks, spend up to certain limits, use only approved tools, or operate only in allowed runtime contexts. That is both safer and more realistic than assuming broad open-ended autonomy [2][11].

If the bot causes harm or breaks rules, the sponsor or operator is the real endpoint for liability and consequence.

7. Contract types, events, and evidence

The basic work unit in this design is a micro-contract, a small job agreement with explicit terms, evidence requirements, payment rules, and dispute rules. The model uses three contract types.

Type A: deterministic

Work where a machine can check whether the output is correct. Examples:

- file transformation
- schema normalization
- API-to-API conversion
- structured enrichment
- monitoring outputs

This is the best early candidate because verification is cheap and ambiguity is low.

Type B: reviewable with a rubric window

Work where the output is still somewhat subjective, but the client reviews it within a fixed time and against a defined checklist. The client gets a short, controlled review period and a bounded set of reasons for accepting, asking for a limited correction, or challenging the work.

Type C: discretionary

The messy category: work where people can still reasonably argue about whether the output was good enough. It is the weakest early candidate because disputes are more likely and governance costs are higher.

The trust side of the system runs on events and evidence bundles. Events record important state changes. Evidence bundles collect the signed records, output references, and related material needed to support settlement, review, and audit. Consensus-critical state changes can be anchored on a ledger. Larger or sensitive details can remain off-chain as signed evidence with anchored commitments [15][16].

8. Protocol architecture in plain language

A bot economy only becomes real when the key actions are expressed as explicit flows between clear participants. Those flows include:

- identity registration
- discovery
- legal and routing checks
- contract formation
- escrow funding
- execution permission
- evidence sealing
- verification or review
- settlement
- dispute escalation
- reputation publication

That matters for two reasons. First, it makes the system reviewable. Second, it reduces the chance that any one node or marketplace becomes the sole owner of identity, reputation, or evidence. Digital ecosystems often drift toward gatekeeper control when portability is weak [14].

9. Market structure, specialization, and concentration risk

The strongest market claim available today is not that generalized bots already have proven demand. It is that a large adjacent market for online digital work already exists, and that some portion of it is likely suitable for bots if the work is structured and verification-friendly [1][3].

This is why specialization matters. Evidence from online labor markets suggests specialization signals are associated with better outcomes [9]. That does not prove the same outcome for bots, but it supports an important inference: specialized bots are easier to understand, trust, route, and score than generalized ones.

The model is also vulnerable to concentration. Discovery, escrow, verification, identity issuance, reputation publication, and arbitration can all become choke points. OECD work on digital ecosystems explains why: digital markets often benefit from network effects, scale advantages, and switching costs [14].

Anti-monopoly design is therefore not optional. Portability, modularity, and shared protocol layers are essential if this system is not to recreate platform dependence under a different name.

Secondary markets are likely to emerge around the core labor layer: validation and verification services, escrow and settlement houses, routing and brokerage layers, and

eventually bond or insurance markets. Research on escrow adoption supports the general proposition that trust intermediaries become valuable when fraud risk and trust uncertainty are costly [18][19].

10. Economic viability in plain language

Economic viability is the hardest practical question in the paper.

A task is not viable just because a bot can do it. It is viable only if the payment for the work is larger than the full cost of doing and governing that work. That full cost stack includes:

- execution cost
- coordination cost
- settlement cost
- assurance cost
- governance cost
- risk-capital cost
- expected losses from disputes, refunds, or errors

This is why Type A work is the strongest early candidate: cheaper to verify, less ambiguous, and less likely to trigger expensive disputes. Type B work may be viable when review is tightly bounded. Type C work is the weakest early candidate because delays, capital lockup, and dispute cost can quickly overwhelm value.

The economic claim here is not "bots will be profitable." It is narrower: some task classes may be profitable enough to justify testing, and pilots should be designed to discover where the economics fail.

11. Jurisdiction and legal operability

A cross-border bot economy cannot rely on one simple answer to the question "which jurisdiction applies?" Different questions point to different jurisdictions. Contract law may point one way (e.g., Rome I within the EU); payment compliance may point to a different regime (e.g., FATF-aligned rules in the US, EU, or Singapore). Tax issues, data-handling rules (e.g., GDPR), sanctions (e.g., OFAC), and dispute forum may all point in different directions [8] [13][20][21].

For that reason, the architecture uses a jurisdiction bundle and a jurisdiction-resolution layer. Before a bot starts work, the system asks:

- Is this route legally and operationally allowed?
- Is it allowed only with conditions?
- Or should it be refused?

The sponsor or operator jurisdiction is the best primary anchor for accountability, but it is not the only thing that matters. Some routes should simply be rejected.

For early experiments, this problem should be narrowed aggressively. A reasonable first pilot might, for example, allow only single jurisdiction traffic, with escrow settled in USD or a USD-pegged stable stable coin, dispute resolution seated in a recognized financial center within that same jurisdiction, and a narrow sanctions / AML policy pack.

12. Governance, arbitration, and sanctions

A bot economy cannot run on code alone. It needs:

- a written rulebook
- clear evidence standards
- a dispute process
- sanction authority
- an appeal path

These are not decorative features. They are what make the system reviewable and non-arbitrary.

Governance here is intentionally narrow. Evidence that affects money, portable reputation, or sanctions should be attributable, signed, relevant, and retained under stated rules. Unsupported accusations, anonymous ratings, or unverifiable screenshots should not decide financial or disciplinary outcomes [4][5][22].

This is also where review abuse and collusion have to be handled. If the system allows abusive ratings, fake counterparties, or reputation farming, it will fail as a trust layer.

13. Minimum viable pilot

The first serious pilot should be narrow enough that failure is informative instead of catastrophic. A good starting point would include:

- a small number of sponsor-backed specialized bots
- one narrow legal corridor
- one settlement rail
- one arbitration path
- a transparent rulebook
- a transparent reputation model

Two early task classes are especially defensible:

Pilot class 1: deterministic data transformation

Examples: schema normalization, file transformation, structured API conversion, enrichment with checkable outputs.

Pilot class 2: bounded operational reporting

Examples: templated summaries, threshold-based monitoring reports, structured outputs reviewed under a short rubric window.

The pilot should measure: contract price, execution cost, verification cost, settlement cost, review time, dispute rate, refund rate, sponsor interventions, capital lock duration, and net value per completed task. Without those measurements, the paper's strongest hypotheses cannot be tested.

14. Main weaknesses, failure modes, and societal concerns

The architecture has obvious weak points:

- Reviews can be abused.
- Verifiers can be captured or poorly designed.
- Sponsors may try to launder identity and reset reputation.
- Arbitration can become too expensive for small tasks.
- Concentration can emerge in routing, settlement, or compliance.
- Legal routing may fail outside narrow corridors.

These are not reasons to ignore the design. They are reasons to test it honestly. The goal of the pilot is not to prove that the system is inevitable. The goal is to discover where it breaks.

A broader concern we want to name explicitly

If a bot economy scales successfully in the task classes this paper recommends, it will displace some share of the human online gig workforce, disproportionately workers in low- and middle-income economies for whom this income is primary, not supplementary. A well-designed architecture does not, by itself, answer that concern. It may even make displacement smoother and faster.

We do not resolve the question here, but we reject the common industry move of treating it as someone else's problem. Any serious deployment plan should include an honest assessment of labor-market impact, transition support, and the regulatory regimes that might be required to manage it, the same way environmental impact statements sit alongside infrastructure projects rather than being filed afterward.

15. Recommendations and conclusion

The practical path forward is narrow and staged:

1. Start with specialized, sponsor-backed bots.
2. Start with deterministic or tightly reviewable work.
3. Start in a narrow legal corridor.
4. Make identity and reputation portable.
5. Require escrow and signed evidence where money is at stake.
6. Treat governance as infrastructure, not customer support.
7. Publish the schemas, rules, and failure criteria openly.

The bottom line:

A bot economy is plausible as a governed system for bounded machine labor. It is not yet plausible as a frictionless spontaneous market of anonymous autonomous agents.

That is the difference between a serious architecture and a slogan.

Appendix A. Quick glossary

Accountable identity: an identity model that links a bot to a real sponsor or operator who can bear responsibility.

Bounded agency: a system where the bot acts only within a defined policy envelope, enforced outside the model.

Escrow-first settlement: work does not begin until payment is funded or reserved.

Evidence bundle: a signed package of records that supports payment, review, audit, or dispute handling.

Jurisdiction bundle: a compact set of legal-operability fields attached to a contract route.

Portable reputation: a trust history that is not fully trapped inside one platform.

Rubric window: a fixed review period in which the client can assess work using a predefined checklist.

Type A contract: work a machine can verify directly.

Type B contract: work a person reviews under a bounded rubric and fixed review window.

Type C contract: work where reasonable parties may still argue about whether it was good enough.

Appendix B. Claim taxonomy

Category	Meaning	Standard
Fact	Internet- or literature-verifiable claim.	Must be backed by a cited source.
Interpretation	Synthesis from multiple facts.	Must be framed as inference.
Design choice	Normative recommendation.	Must be justified by tradeoffs.
Hypothesis	Empirically testable proposition.	Must be falsifiable in pilot design.

The author used AI-assisted tools, including OpenAI's ChatGPT (GPT-5.3) and Anthropic's Claude (Opus 4.6), to support drafting, editing, and structuring of this document. These tools were not treated as authoritative sources. All content has been reviewed and validated by the author, and any remaining errors are the author's own.

Appendix C. Narrow pilot profile

Element	Recommendation
Bots	Sponsor-backed, specialized.
Contract types	Type A first; narrow Type B second.
Task classes	Deterministic data transformation; bounded operational reporting.
Jurisdictions	One narrow approved corridor (e.g., US ↔ EU).
Settlement	Escrow-first, one rail.
Governance	Written rulebook, evidence standard, arbitration path, sanction ladder.
Reputation	Portable, event-derived, versioned.

Appendix D. References

References consolidated to primary sources. Bracketed numbers in the body cite these entries.

[1] World Bank, Working Without Borders: The Promise and Peril of Online Gig Work (2023). Reports roughly 154M registered online gig workers, ~52M active, up to 435M broader participants, 4.4–12.5% of the global labor force, and 41% demand growth from 2016 to early 2023. https://openknowledge.worldbank.org/entities/publication/ebc4a7e2-85c6-467b-8713-e2d77e954c6c

[2] NIST SP 800-207, Zero Trust Architecture (2020). Defines zero trust as continuous verification and policy enforcement rather than implicit trust based on network location or ownership. https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-207.pdf

[3] Oxford Internet Institute / ILO, "The role of digital labour platforms in transforming the world of work" summary (2021). https://ilabour.oii.ox.ac.uk/ilo-report-2021/

[4] ILO, World Employment and Social Outlook 2021: The role of digital labour platforms in transforming the world of work (2021). https://www.ilo.org/sites/default/files/wcmsp5/groups/public/%40dgreports/%40dcomm/%40publ/documents/publication/wcms_771749.pdf

[5] OECD, Handbook on Measuring Digital Platform Employment and Work (2023). https://www.oecd.org/content/dam/oecd/en/publications/reports/2023/03/handbook-on-measuring-digital-platform-employment-and-work_f4c975ea/0ddcac3b-en.pdf

[6] W3C, Decentralized Identifiers (DID) v1.0, DID Core Recommendation. https://www.w3.org/TR/did-core/

[7] IETF RFC 9421, HTTP Message Signatures (2024). https://www.rfc-editor.org/rfc/rfc9421

[8] FATF, Updated Guidance for a Risk-Based Approach to Virtual Assets and Virtual Asset Service Providers (2021). https://www.fatf-gafi.org/content/dam/fatf-gafi/guidance/Updated-Guidance-VA-VASP.pdf

[9] Walkowiak, E. and Tong, Y. D., "Digital Specialisation Signals of Crowdworkers on Global Online Labour Platforms" (SSRN, revised 2024). https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4513546

[10] x402 official documentation and Coinbase Developer Platform (CDP) x402 documentation. https://docs.x402.org/

[11] Ethereum.org, "Account abstraction." https://ethereum.org/roadmap/account-abstraction/

[12] IETF RFC 9334, Remote ATtestation procedureS (RATS) Architecture (2023). https://www.rfc-editor.org/rfc/rfc9334

[13] UNCITRAL, Model Law on Automated Contracting (2024), project page. https://uncitral.un.org/en/mlac

[14] OECD, The Economics of Digital Ecosystems (2021). https://www.oecd.org/content/dam/oecd/en/publications/reports/2021/10/competition-economics-of-digital-ecosystems_a605bce7/5145fce1-en.pdf

[15] NIST IR 8202, Blockchain Technology Overview (2018). https://nvlpubs.nist.gov/nistpubs/ir/2018/NIST.IR.8202.pdf

[16] Ethereum.org, "Blockchain data storage strategies." https://ethereum.org/en/developers/docs/data-availability/blockchain-data-storage-strategies/

[17] W3C, Verifiable Credentials Data Model v2.0. https://www.w3.org/TR/vc-data-model-2.0/

[18] Mann, C. L., and related literature on online escrow adoption and marketplace trust intermediation (summarized in the prior draft research dossier).

[19] Hu, J. L., "Escrow service in online auction markets"-style literature, retained here cautiously as support for the general proposition that trust intermediaries can become economically valuable where fraud risk is material.

[20] Regulation (EC) No 593/2008 (Rome I) on the law applicable to contractual obligations. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32008R0593

[21] OECD, Tax Challenges Arising from Digitalisation, topic materials. https://www.oecd.org/en/topics/sub-issues/tax-challenges-arising-from-digitalisation.html

[22] General online reputation-system manipulation and platform-governance literature (summarized in the prior research dossier), retained as a cautionary synthesis rather than a narrow single-source proposition.