Can AI Be Decentralized and Deindustrialized

August 17, 2026

Table of Contents

By https://app.undermind.ai/

Can AI Be Decentralized and Deindustrialized

Direct answer

Yes, but the most realistic goal is not to eliminate industry or make every municipality technologically self-sufficient. It is to replace hyperscale dependence with a **locally autonomous, globally federated**

system in which most tasks use small local models, communities share compute directly, and difficult hardware and training tasks are coordinated through public or commons institutions.

The 2020s research supports five linked propositions:

- Many useful AI tasks do not require frontier-scale models.
- Small language models can run locally on modest hardware when they are trained and evaluated for specific tasks [Kan24, Lu25].
- Groups of devices can pool resources through federated, hierarchical, gossip, and peer-to-peer learning [Yua23, Hu22, Fra23].
- Decentralization can reduce data movement and cloud dependence, but it does not automatically reduce energy use [Sav21].
- Technical distribution becomes durable autonomy only when communities control the hardware, software, data, maintenance, and rules of interconnection [Ros20, Sch22, Ter23].

The strongest answer to the original query is therefore:

> AI can be deindustrialized at the level of many applications, but not completely at the level of frontier-model training. It can become genuinely decentralized when right-sized models, shared peer-to-peer compute, open hardware, repairable devices, and democratic confederation are designed as one system.

What “decentralized” should mean

The word can refer to different layers of the system. A decentralized network is not necessarily a decentralized economy.

Layer	Hyperscale arrangement	Decentralized alternative
Tasks	One general model used for everything	Models sized for particular tasks and contexts
Inference	Remote API call to a cloud provider	Local or nearby inference, with escalation only when needed
Training data	Data pooled by a central firm	Data stays with households, institutions, or communities
Coordination	Central parameter server	Gossip, federated, hierarchical, or peer-to-peer coordination
Compute ownership	Corporate cloud infrastructure	Municipal, cooperative, public, or commons compute
Hardware	Proprietary devices with short lifecycles	Open, modular, repairable, and interoperable hardware
Governance	Platform rules imposed from above	Local self-government plus confederal coordination

Layer	Hyperscale arrangement	Decentralized alternative
Connectivity	Global platform as the default path	Local services and caches connected through federated networks

This distinction prevents a common mistake. A peer-to-peer protocol can still run on corporate servers. An open model can still depend on proprietary GPUs. A local device can still contain chips whose design and manufacture are controlled elsewhere. The political-economy literature describes AI as a stack of infrastructure, models, applications, labor, and capital rather than as software alone [Vli24, Val24].

The first step is right-sizing the task

The strongest case for deindustrialized AI is not frontier training. It is replacing the assumption that every task requires a large general-purpose model.

Small language models can be compressed through quantization, pruning, distillation, and parameter-efficient adaptation. This makes local inference possible on phones, single-board computers, and other edge devices [Xu22, Don24].

TinyLLM developed models between 30 and 124 million parameters for sensing tasks and ran them on single-board computers. On an Orange Pi Zero 2W, a device unable to run larger models, a custom model generated roughly six tokens per second [Kan24]. These models were useful for gesture detection, localization, and sensor classification, but they were not general replacements for large language models.

A wider evaluation of small language models finds that models between roughly 100 million and 5 billion parameters can be highly competitive on particular tasks. Some 2–4 billion parameter models perform close to or better than larger models after fine-tuning, especially in domain-specific tool use [Lu25]. They remain weaker at open-ended reasoning, broad knowledge, and some forms of in-context learning.

This supports a **local-first model hierarchy**:

1. Run a small model on the local device whenever it can meet the task requirement.
2. Use a nearby community or municipal model when the local model is insufficient.
3. Federate with other local models when shared learning is useful.
4. Send only genuinely difficult or rare tasks to a regional or global model.

Such a hierarchy would reduce routine dependence on cloud APIs without pretending that every task can be handled locally. It would also make model choice a political and ecological decision rather than a race toward maximum parameter count.

Sharing compute peer to peer

The original idea of clustering modest servers has several technical forms.

Federated learning

Federated learning lets devices or institutions train a shared model without sending raw data to a central repository. Conventional federated learning usually retains a central server for aggregation, but decentralized federated learning replaces or reduces that role through gossip, mesh, ring, star, and hybrid topologies [Yua23].

Hierarchical clusters

Devices can form local clusters, aggregate updates within the cluster, and periodically exchange summaries with other clusters. Spread demonstrates this arrangement by assigning some edge devices to act as cluster heads. It achieved an 8.05-fold training speedup over a conventional parameter-server design and a 5.58-fold speedup over an all-reduce system [Hu22].

The limitation is important: Spread still retained a cloud coordinator. It decentralized a bottleneck rather than eliminating central infrastructure.

Fully peer-to-peer learning

Serverless collaborative learning distributes the server function across an aggregation committee. Secure multiparty computation and verifiable sharing can prevent a single party from inspecting private updates or corrupting the model [Fra23]. Other peer-to-peer systems use reputation and co-utility mechanisms so that honest computation becomes the most advantageous strategy for participating nodes [Dom20].

Volunteer and idle-resource computing

A confederation could also pool computers that are not continuously dedicated to AI. Volunteer-computing research shows that large numbers of consumer devices can provide substantial aggregate capacity, but only with scheduling systems that account for hardware differences, memory, user preferences, availability, and project requirements [And21].

The best workloads for this model are:

- batch jobs that can be divided into independent pieces
- local inference requests routed to nearby nodes
- model evaluation and testing
- data preprocessing
- fine-tuning of compact models
- federated learning across institutions
- caching and serving local knowledge

The worst workloads are tightly synchronized frontier-model training and any task that requires every node to exchange large tensors at every step.

Locally autonomous but globally networked

Decentralization does not require isolation or autarky. The more useful model is a federation of autonomous communities that share standards, designs, model improvements, and selected compute resources.

Research on Indigenous shared networks in Mexico offers a concrete analogue. Communities build and maintain their own first-mile infrastructure while connecting to the global Internet through selective external links. They also operate local intranets and caches so that education and cultural material remains available locally rather than requiring constant access to distant commercial services [Ros21].

The same principle could apply to AI:

- A community keeps its sensitive data and routine services local.
- Local servers cache models, documents, and educational resources.
- Communities exchange model updates or openly licensed improvements.
- Regional federations provide difficult services and technical support.
- Global networks share blueprints, protocols, safety findings, and repair knowledge.

This is not a retreat from the Internet. It is a change in the relationship between local systems and the global network: global connectivity becomes an interconnection between autonomous communities rather than a permanent dependency on a few platforms.

Research on peer-to-peer and information commons warns that such systems are fragile if they lack maintenance, legal protection, funding, and clear responsibility [Ros20]. Community networks also show that local ownership and technical knowledge are as important as physical connectivity [Lyn21].

How the hardware could be produced

A decentralized confederation should not try to make every component in every locality. That would be inefficient and would reproduce a different kind of compulsion toward industrial scale. The better model is **confederated specialization**: local control over use, repair, and deployment, combined with regional and global cooperation for components that cannot reasonably be made everywhere.

A four-level hardware system

Level	Capabilities	Political purpose
Local	Repair, reuse, enclosures, cooling, racks, sensors, device assembly	Keep maintenance and practical knowledge in the community
Municipal or regional	PCB assembly, testing, fabrication labs, replacement parts, low-power servers	Build and maintain common infrastructure close to users

Level	Capabilities	Political purpose
Confederated	Shared component libraries, procurement, training, standards, logistics, specialized board production	Prevent isolated communities from reinventing everything
Global	Advanced chips, semiconductor fabrication, specialized tools, rare materials	Coordinate scarce capabilities without allowing private enclosure

Open-source hardware research shows that communities can decentralize early design, prototyping, documentation, and modification. It also shows that full decentralization from requirements through manufacture remains difficult because hardware requires expensive tools, testing, standards, and physical supply chains [Bon21b].

The practical design principles are therefore crucial:

- modular components that can be replaced independently
- reversible fasteners instead of sealed assemblies
- open interfaces and documented specifications
- standardized parts and connectors
- long software support periods
- repair and upgrade access
- local diagnostic tools
- designs that can be manufactured with common CNC, laser-cutting, or 3D-printing processes
- shared digital version control for hardware designs [Bon21b, Poh21]

Open hardware can also reach below the level of finished devices. Open instruction-set architectures such as RISC-V, open electronic-design tools, and open process documentation have created pathways toward community-accessible chip design. Open-source silicon initiatives have demonstrated low-volume manufacturing through shared design and fabrication arrangements, although advanced semiconductor production remains highly specialized [Han21].

The aim is not to make a complete advanced GPU in every town. It is to ensure that communities are not locked into one vendor for every layer of the system. Open interfaces, repairable boards, replaceable accelerators, and multiple manufacturing partners can make the hardware stack more politically contestable.

Hardware in a municipalist or confederal society

A democratic-confederal model would need institutions that connect local autonomy to wider coordination.

Local assemblies

Local communities would decide:

- which services should exist
- which data may leave the community
- what energy and material budgets apply
- whether AI is appropriate for a given task
- how local compute is shared
- who is responsible for repair and maintenance

A local assembly should not need permission from a central platform to change its model, hardware, or data policy.

Confederated technical councils

Communities would delegate narrowly defined functions upward:

- interoperability standards
- shared security updates
- model and hardware registries
- pooled procurement
- training and technical education
- regional repair and fabrication capacity
- coordination of scarce compute
- support for communities with fewer resources

Delegation should be revocable and task-specific. The confederation would coordinate capabilities without becoming the owner of every local system.

Global design commons

A global commons could maintain:

- open hardware designs
- model weights and training methods
- public datasets with community-defined access rules
- evaluation suites
- documentation and repair manuals
- safety and privacy protocols
- energy and lifecycle benchmarks

The principle would be “design global, manufacture and govern locally.” Post-growth research describes this kind of arrangement as combining collaborative commons, assisted administration, peer production, and egalitarian exchange while keeping communities locally rooted and globally connected [Isa26].

Financing and ownership

Local autonomy cannot depend on unpaid enthusiasm alone. Community infrastructure is capital-intensive and requires long-term technical capacity, funding, and maintenance [Wee25].

Possible institutions include:

- municipal compute utilities
- cooperative server farms
- regional equipment pools
- public procurement for open hardware
- shared repair and fabrication centers
- cooperative development funds
- perpetual-purpose trusts
- federated reserves funded by participating communities

Research on community-owned platforms proposes flexible cooperative incorporation, public lending and insurance, preferred public procurement, interoperability requirements, and pooled reserves for future cooperative development [Sch21]. These mechanisms matter because ownership is not created by open code alone.

Ecological limits

Decentralization can reduce some forms of ecological harm, but it is not automatically green.

A comparison of centralized, federated, and decentralized learning finds that federated learning can reduce end-to-end energy use by 30–40 percent in low-power, low-bandwidth Internet of Things settings. It can also shift computation toward regions with lower-carbon electricity [Sav21].

But uneven local data can double the number of training rounds and increase total energy use by up to 40 percent. Repeated communication, duplicated models, inefficient devices, and short hardware lifecycles can erase the benefit [Sav21].

The full lifecycle includes:

- mineral extraction
- chip and board production
- electricity and cooling
- network traffic
- device replacement
- repair and logistics
- data labor
- e-waste

AI's supply chain already distributes these burdens across mining regions, manufacturing centers, data-centre locations, and e-waste sites [Val24]. Decentralization should therefore be evaluated by **energy and material use per useful social task**, not by the number of servers or the absence of a central cloud.

A post-growth system would also need a sufficiency rule: use the least capable model that meets the need, avoid always-on computation, cache common results locally, schedule intensive jobs when renewable power is available, and refuse applications whose social value does not justify their material cost.

What remains industrial

The global compute geography shows that frontier training hardware is concentrated in a small number of countries and hyperscale cloud regions. Many countries have compute suitable mainly for inference, while most have little or no public cloud AI infrastructure [Leh24b].

This means that a confederated system should not pretend that advanced fabrication and frontier training can be made fully local in the near term. Some capabilities will remain globally specialized. The political goal is to prevent specialization from becoming private dependence.

Several claims should therefore be rejected:

- A pile of small servers does not automatically equal a supercomputer; memory and network latency matter.
- Federated learning is not necessarily decentralized if one company still controls the aggregator.
- Open hardware does not eliminate dependence on mines, fabs, or specialized machinery.
- Local AI is not automatically low-carbon.
- A global federation can become centralized if standards, financing, or maintenance are controlled by one institution.
- Volunteer participation can become hidden exploitation if maintenance is not treated as socially necessary work.

A practical blueprint

A plausible decentralized AI system would work as follows:

1. **Local-first deployment:** Most routine inference happens on small models owned and operated by households, associations, clinics, schools, workshops, and municipalities.
2. **Task-based model selection:** A community chooses a model according to accuracy, latency, energy, privacy, and maintainability rather than prestige or parameter count.
3. **Community compute pools:** Nearby devices and servers share idle capacity through a scheduler that respects energy budgets, hardware limits, and member preferences.
4. **Federated learning:** Sensitive data remain local while model improvements circulate through peer-to-peer or hierarchical protocols.
5. **Regional escalation:** More demanding tasks are sent to a cooperative or public regional cluster, not automatically to a corporate cloud.
6. **Global federation:** Communities share open designs, model improvements, safety research, and selected compute across long-distance links.
7. **Open hardware:** Devices use documented interfaces, repairable modules, open software, and multiple suppliers wherever possible.
8. **Confederal governance:** Local assemblies govern local use; higher bodies coordinate only the functions that genuinely require wider scale.
9. **Material accounting:** Every layer reports energy, water, hardware, labor, and e-waste costs.
10. **Right of exit:** Communities retain the ability to fork software, replace components, leave a federation, or build a compatible alternative.

This is not a fully demonstrated social system. It is a synthesis of technical research on edge and decentralized learning with research on community networks, open hardware, commons governance, and post-growth infrastructure [Yua23, Ros21, Bon21b, Sch22, Isa26].

Conclusion

The original query is directionally right that AI is being made more centralized and resource-intensive than every use case requires. The answer is not that all AI can be reduced to a few home computers. The answer is to break the assumption that one industrial model must sit behind every task.

The most promising path is a **federated ecology of AI**:

- small models for ordinary tasks
- local data and local inference
- peer-to-peer sharing of compute
- regional public or cooperative clusters
- globally shared designs and protocols
- repairable and open hardware
- democratic confederation rather than platform dependence

The key distinction is between **autarky** and **autonomy**. Autarky would require every locality to reproduce the entire industrial stack. Autonomy requires the ability to govern local systems, change providers, maintain essential capabilities, and enter wider relations without surrendering control.

The decisive political question is therefore not whether every community can manufacture frontier GPUs. It is whether communities can collectively control enough of the AI stack—models, data, local compute, repair, standards, and interconnection—to make large-scale industry a tool they can use selectively rather than a private infrastructure on which they must permanently depend.

References

[Kan24] S. V. Kandala, P. Medaranga, and A. Varshney, “TinyLLM: A Framework for Training and Deploying Language Models at the Edge Computers,” *ArXiv*, vol. abs/2412.15304, Dec. 2024, doi: 10.48550/arXiv.2412.15304.

[Lu25] Z. Lu *et al.*, “Demystifying Small Language Models for Edge Deployment,” *Annual Meeting of the Association for Computational Linguistics*, pp. 14747–14764, 2025, doi: 10.18653/v1/2025.acl-long.718.

[Yua23] L. Yuan, L. Sun, P. Yu, and Z. Wang, “Decentralized Federated Learning: A Survey and Perspective,” *IEEE Internet of Things Journal*, vol. 11, pp. 34617–34638, Jun. 2023, doi: 10.1109/JIOT.2024.3407584.

[Hu22] C. Hu, H. Liang, X. Han, B. Liu, D. Cheng, and D. Wang, *Spread: Decentralized Model Aggregation for Scalable Federated Learning*. 2022. doi: 10.1145/3545008.3545030.

[Fra23] O. Franzese *et al.*, “Robust and Actively Secure Serverless Collaborative Learning,” *ArXiv*, vol. abs/2310.16678, Oct. 2023, doi: 10.48550/arXiv.2310.16678.

[Sav21] S. Savazzi, S. Kianoush, V. Rampa, and M. Bennis, “A framework for energy and carbon footprint analysis of distributed and federated edge learning,” *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1564–1569, Mar. 2021, doi: 10.1109/PIMRC50174.2021.9569307.

[Ros20] M. D. Rosnay and F. Musiani, “Alternatives for the Internet: A Journey into Decentralised Network Architectures and Information Commons,” Aug. 19, 2020. doi: 10.31269/triplec.v18i2.1201.

[Sch22] N. Schneider, “Governable Stacks against Digital Colonialism,” Jan. 12, 2022. doi: 10.31269/triplec.v20i1.1281.

[Ter23] P. A. Terzis, “Building programmable commons,” *Law, Innovation and Technology*, vol. 15, pp. 583–616, Jul. 2023, doi: 10.1080/17579961.2023.2245676.

[Vli24] F. N. van der Vlist, A. Helmond, and F. Ferrari, “Big AI: Cloud infrastructure dependence and the industrialisation of artificial intelligence,” *Big Data & Society*, vol. 11, Mar. 2024, doi: 10.1177/20539517241232630.

[Val24] A. Valdivia, “The supply chain capitalism of AI: a call to (re)think algorithmic harms and resistance through environmental lens,” *Information, Communication & Society*, vol. 28, pp. 2118–2134, Oct. 2024, doi: 10.1080/1369118X.2024.2420021.

[Xu22] C. Xu and J. McAuley, “A Survey on Model Compression and Acceleration for Pretrained Language Models,” *AAAI Conference on Artificial Intelligence*, pp. 10566–10575, Feb. 2022, doi: 10.1609/aaai.v37i9.26255.

[Don24] Y. Dong, X. Fan, F. Wang, C. Li, V. C. M. Leung, and X. Hu, “Fine-Tuning and Deploying Large Language Models Over Edges: Issues and Approaches,” *IEEE Communications Magazine*, vol. 64, pp. 94–101, Aug. 2024, doi: 10.1109/MCOM.001.2400602.

[Dom20] J. Domingo-Ferrer, A. Blanco-Justicia, D. Sánchez, and N. Jebreel, “Co-Utile Peer-to-Peer Decentralized Computing,” *2020 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGRID)*, pp. 31–40, May 2020, doi: 10.1109/CCGrid49817.2020.00-90.

[And21] D. P. Anderson, “Globally Scheduling Volunteer Computing,” *Future Internet*, vol. 13, p. 229, Aug. 2021, doi: 10.3390/fi13090229.

[Ros21] F. R. Rosa, “From community networks to shared networks: the paths of Latin-Centric Indigenous networks to a pluriversal internet,” Sep. 15, 2021. doi: 10.1080/1369118X.2022.2085614.

[Lyn21] C. R. Lynch, “Internet infrastructure and the commons: grassroots knowledge sharing in Barcelona,” Feb. 03, 2021. doi: 10.1080/00343404.2020.1869200.

[Bon21b] J. Bonvoisin, R. Mies, and J. Boujut, “Seven observations and research questions about Open Design and Open Source Hardware,” Oct. 19, 2021. doi: 10.1017/dsj.2021.14.

[Poh21] J. Pohl, A. Höfner, E. Albers, and F. Rohde, “Design Options for Long-lasting, Efficient and Open Hardware and Software,” Feb. 11, 2021. doi: 10.14512/OEWO360120.

[Han21] F. Hannig and J. Teich, “Open Source Hardware,” *Computer*, vol. 54, pp. 111–115, Oct. 2021, doi: 10.1109/mc.2021.3099046.

[Isa26] Å. Isacson, M. Adelfio, and L. Thuvander, “Post-growth technologies: a scoping review of innovations related to degrowth, sharing economy and self-sufficiency,” *SN Social Sciences*, vol. 6, Feb. 2026, doi: 10.1007/s43545-026-01329-4.

[Wee25] D. A. Weeden, D. Kelly, D. Breen, and S. College, “Sisyphus’s Broadband: Exploring models of rural community participation in digital infrastructure and connectivity,” *J. Community Informatics*, vol. 21, Feb. 2025, doi: 10.15353/joci.v21i1.5508.

[Sch21] N. Schneider, “Enabling Community-Owned Platforms: A Proposal for a Tech New Deal,” Dec. 03, 2021. doi: 10.31219/osf.io/cp459.

[Leh24b] V. Lehdonvirta, B. Wu, and Z. Hawkins, “Compute North vs. Compute South: The Uneven Possibilities of Compute-based AI Governance Around the Globe,” *AAAI/ACM Conference on AI, Ethics, and Society*, pp. 828–838, Oct. 2024, doi: 10.1609/aies.v7i1.31683.