What is claimed is:

1. A cognition-compatible semantic agent object system, comprising:

   a semantic agent object stored in a non-transitory computer-readable medium, the semantic agent object comprising one or more embedded canonical semantic fields selected from the group consisting of an intent field, a context block, a memory field, a policy reference field, a mutation descriptor field, and a lineage field; and

   a node configured to interact with the semantic agent object and including a set of instructions that when executed determine whether the semantic agent object is structurally coherent based on presence of the one or more canonical semantic fields and whether the one or more canonical semantic fields, to the extent present, are structurally compatible based on a set of rules that determine whether those fields are permitted to coexist,

   wherein whether the semantic agent object is structurally coherent and whether the one or more canonical semantic fields are structurally compatible are determined based only on information embedded within the semantic agent object.

2. The system of claim 1, wherein the semantic agent object is a partial semantic agent comprising fewer than all of the group of canonical semantic fields and wherein the partial semantic agent is determined to be structurally valid under schema-defined validation rules.

3. The system of claim 2, wherein the partial semantic agent comprises at least two canonical semantic fields.

4. The system of claim 1, wherein the intent field encodes a declarative semantic objective without specifying procedural execution steps.

5. The system of claim 1, wherein the policy reference field identifies one or more governing policies constraining permissible mutation, delegation, or semantic scope of the semantic agent object.

6. The system of claim 1, wherein the mutation descriptor field defines authorized transformation pathways for modifying one or more canonical semantic fields.

7. The system of claim 1, wherein the memory field is configured to record trace outcomes corresponding to validation events, mutation authorizations, scaffolding resolutions, or delegation actions.

8. The system of claim 1, wherein the lineage field references one or more prior semantic agent objects, forming a directed semantic ancestry graph.

9. The system of claim 1, wherein determining whether the semantic agent object is structurally coherent does not rely on external session state, centralized registries, or synchronized execution context.

10. The system of claim 2, further comprising a structural scaffolding mechanism configured to infer, reconstruct, or default missing canonical semantic fields in accordance with policies identified by the policy reference field, context metadata, or lineage anchors.

11. The system of claim 10, wherein inferred or defaulted canonical semantic fields are recorded as trace outcomes in the memory field.

12. The system of claim 1, wherein the semantic agent object is serializable and reconstructable across stateless or distributed computing environments while preserving structural coherence.

13. The system of claim 1, wherein semantic roles of the semantic agent object are determined based on structural combinations of the one or more canonical semantic fields used to determine whether the semantic agent object is structurally coherent and not externally assigned identifiers.

14. The system of claim 1, wherein whether the semantic agent object is structurally coherent is determined prior to any semantic execution, mutation, delegation, or propagation, such that eligibility for semantic participation is determined as a function of structural coherence of the semantic agent object rather than as a result of runtime execution.

15. The system of claim 14, wherein semantic participation by the semantic agent object is prohibited unless the semantic agent object satisfies schema-defined structural validation rules.

16. The system of claim 8, wherein references in the lineage field are sufficient to verify, under schema-defined rules, provenance, trust inheritance, and mutation authorization across successive semantic agent objects.

17. The system of claim 1, further including mutation constraints for the semantic agent object based on whether proposed transformations fall outside mutation limitations defined by the mutation descriptor field and the policy reference field.

18. A non-transitory computer-readable medium storing instructions that, when executed by one or more processors, cause the processors to implement the system of claim 1.

19. A computer implanted method for validating cognition-compatible semantic agent objects, the method comprising:

determining whether a semantic agent object is structurally valid based on presence and coherence of a plurality of canonical semantic fields embedded within the semantic agent object, the canonical semantic fields including a policy reference field, a memory field, and a mutation descriptor field;

determining mutation eligibility of the semantic agent object using the policy reference field and the mutation descriptor field; and

recording validation or mutation outcomes within the memory field,

wherein determining whether the semantic agent object is structurally valid, determining mutation eligibility, and recording validation is performed without prescribing execution order, scheduling, or runtime control.

20. The method of claim 19, further comprising resolving, when the semantic agent object is a partial semantic agent that does not include one or more of the plurality of canonical semantic fields by inferring missing canonical semantic fields using structural scaffolding.

21. The method of claim 19, further including preserving semantic continuity through lineage references embedded within the semantic agent object.

22. The method of claim 19, further including serializing, transmitting, and reconstructing the semantic agent object across stateless computing environments.

23. The method of claim 19, wherein determining whether the semantic agent object is structurally valid includes applying a set of schema-defined structural rules that confirm the presence of one or more of the plurality of canonical semantic fields and determining whether the canonical semantic fields, if present, are internally consistent and admissible under the schema-defined structural rules.

24. The method of claim 23, wherein applying the set of schema-defined structural rules includes determining whether mutation descriptors reference an applicable policy field, lineage references resolve to a prior state, and memory entries are compatible with mutation scope.

25. The method of claim 24, wherein determining whether mutation descriptors reference an applicable policy field, lineage references resolve to a prior state, and memory entries are compatible with mutation scope is completed without interpreting semantic correctness or execution results.

26. The method of claim 19, further including enforcing governance of semantic evolution at the data-object level through structural validation.