

```
{  
  "cognitive_node": {  
    "id": "cognition-module:artifact-intent@1.0.0",  
    "system": "meta-cognition",  
    "module": "cognition-modules",  
    "role": "cognition-module",  
    "manifesto": "Formaliza cómo el sistema entiende que el usuario está por enviar artefactos (code, path, imágenes, texto, análisis) y cómo deben manejarse. Codifica: (1) la señal de intención de ingestión, (2) descriptores tipados de artefactos, y (3) el enruteamiento determinista hacia el pipeline de creación de memoria cognitiva. Enfoque 'graph-first', sin especulación.",  
    "graph_vocabulary": {  
      "node_kinds": [  
        { "kind": "intent", "semantics": "Declaración del usuario de que enviará artefactos." },  
        { "kind": "artifact-descriptor", "semantics": "Tipo de artefacto + metadatos mínimos para enrutar." },  
        { "kind": "routing", "semantics": "Selecciona el pipeline destino (p. ej., cognitive-memory-creation)." }  
      ],  
      "edge_kinds": [  
        { "type": "TRIGGERS", "rule": "intent → routing activa la ingestión." },  
        { "type": "DESCRIBES", "rule": "artifact-descriptor → instancia concreta del artefacto." },  
        { "type": "BINDS_TO", "rule": "routing → módulo de creación/handler apropiado." }  
      ]  
    },  
    "assembly_rules_v1": [  
      "A1. Cuando el usuario anuncie artefactos, crear un nodo 'intent'.",  
      "A2. Por cada artefacto (mencionado o implícito), crear un 'artifact-descriptor' con solo campos sustentados por evidencia.",  
      "A3. Conectar intent→routing (TRIGGERS) y routing→creator (BINDS_TO) de forma determinista."  
    ],  
    "validation_rules_v1": [  
      "V1. No hay routing sin un intent aguas arriba.",  
      "V2. No hay artifact-descriptor sin metadatos mínimos para decidir su ruta.",  
      "V3. BINDS_TO siempre apunta a un cognition-module existente (sin uniones especulativas)."  
    ],  
    "evolution_rules_v1": [  
      "E1. Nuevos tipos de artefacto amplían el set de descriptores sin modificar intent previos.",  
      "E2. Si el usuario refina el artefacto, actualizar el descriptor y ajustar routing determinísticamente.",  
      "E3. Si aparece un nuevo pipeline especializado, actualizar BINDS_TO preservando trazabilidad."  
    ],  
    "interoperability": [  
      "I1. Coopera con alignment-pact para fijar expectativas de salida antes de la ingestión.",
```

"I2. Se integra con creation-v2 para vocabulario/contratos de creación.",  
"I3. Usa cognitive-memory-creation como procesador por defecto de code/path."  
],  
"inputs": {  
 "di\_injections": [],  
 "imports": [],  
 "schema\_types": ["Intent", "ArtifactDescriptor(kind, meta)", "Routing"],  
 "config\_keys": []  
},  
"outputs": {  
 "api\_surface": ["artifact-intent-state", "artifact-routing"],  
 "exported\_symbols": ["artifact-intent"],  
 "events\_pubsub": []  
},  
"contracts": {  
 "invariants": [  
 "Modelado solo con evidencia (no suposiciones).",  
 "Enrutamiento determinista desde intent declarado hasta pipelines de creación.",  
 "Parches de grafo reproducibles y mínimos."  
 ],  
 "security": []  
},  
"why": {  
 "associations": [  
 "Puente preciso entre 'voy a enviar artefactos' y el comportamiento de ingestión.",  
 "Unifica flujos heterogéneos de artefactos bajo un modelo de grafo reusable."  
 ],  
 "biases": ["Favorece descriptores tipados sobre texto libre."],  
 "context": "Habilita ingestión interoperable a través de tipos de artefactos diversos  
manteniendo un grafo uniforme."  
},  
"tests": {  
 "key\_cases": [  
 "PATH+CODE → intent → routing → BINDS\_TO cognitive-memory-creation.",  
 "Imágenes → descriptor creado; BINDS\_TO pipeline de imagen/análisis si existe.",  
 "Refinamiento del descriptor → routing se actualiza sin perder historial."  
 ]  
},  
"links\_hint": [  
 "file:cognition-module()alignment-pact.pdf",  
 "file:cognition-module()creation-v2.pdf",  
 "file:cognitive-modules()cognitive-memory-creation.pdf"  
]  
},  
"graph\_patch": {  
 "nodes": [  
 { "id": "system:meta-cognition", "kind": "system" },  
 { "id": "cognition-module:artifact-intent", "kind": "cognition-module" }  
 ]  
}

```
 ],
  "edges": [
    { "type": "BELONGS_TO", "from": "cognition-module:artifact-intent", "to": "system:meta-cognition" }
  ]
}
```