

 MÓDULO 04

Memória

O Segredo que Ninguém Ensina

Arquitetura de memória estruturada: como fazer seu agente lembrar, aprender e evoluir entre sessões

DURAÇÃO

25 min

FORMATO

Demo + Diagramas

KIT

PRD + 7 Templates

O Problema: Alzheimer Digital

Toda vez que uma sessão termina, seu agente **esquece tudo**. Decisões que você tomou ontem? Esqueceu. Preferências que você explicou 3 vezes? Esqueceu. Projetos em andamento? Esqueceu.

Sem memória estruturada, você repete contexto todo santo dia. É como contratar um assistente brilhante que acorda com amnésia toda manhã.

⚠ Caso real — Dia 2

Sessão estourou 173k+ tokens (limite 160k). Agente parou de responder. Causa: zero compactação configurada, contexto crescendo infinitamente. Fix: configurar compaction + reserveTokensFloor ANTES de precisar.

A Solução: Memória em Camadas

Em vez de depender só do contexto da sessão, criamos uma arquitetura de 4 camadas — do efêmero ao permanente:

Sessão

Contexto atual

→

Nota Diária

memory/YYYY-MM-DD.md

→

Topic Files

decisions, lessons...

→

MEMORY.md

Índice central

Cada camada filtra e destila. A sessão é raw. A nota diária é registro. Os topic files são curadoria. O MEMORY.md é sabedoria destilada.

Topic Files Especializados

Dentro de `memory/`, cada arquivo tem um propósito claro:

decisions.md — Decisões Permanentes

Tudo que foi decidido e NÃO deve ser revertido sem discussão. "Todas credenciais no 1Password", "Sonnet pra crons", "Hub model > Mesh model". Nunca perder.

lessons.md — Lições Aprendidas

Erros, padrões, insights. Com retenção inteligente: Estratégicas = permanentes (padrões, filosofia). Táticas = expiram em 30 dias (bugs, workarounds). Revisão mensal.

projects.md — Projetos Ativos

Estado atual de cada projeto: o que tá em andamento, o que tá parado, o que foi concluído. O agente consulta antes de sugerir trabalho.

people.md — Pessoas & Equipe

Contatos, parceiros, equipe. Quem faz o quê, como se comunicar, preferências de cada pessoa. Essencial pra multi-agentes.

pending.md — Aguardando Input

Tudo que depende de você (humano). O agente não esquece — e te cobra. "Bruno precisa fornecer API key da Anthropic", "Aguardando decisão sobre pricing".

MEMORY.md como Índice

O `MEMORY.md` na raiz do workspace é o **índice** — aponta pros topic files sem duplicar conteúdo. É o "sumário executivo" da memória do agente. Carregado a cada sessão.

Compactação & A Regra Inviolável

Configurando a Compactação

Sem compactação, sua sessão cresce até estourar. A config essencial:

⚙️ Config de Compactação

```
{  
  "compaction": { "mode": "default" },  
  "contextTokens": 160000,  
  "reserveTokensFloor": 30000  
}
```

O `reserveTokensFloor` garante que o agente termina o raciocínio antes de compactar — sem ele, respostas ficam cortadas no meio.

☐ REGRA INVOLÁVEL

Antes de CADA compactação, o agente DEVE extrair: lições → lessons.md, decisões → decisions.md, pendências → pending.md. Se não extrair antes de compactar, **perde 80% do valor**

. É como formatar o HD sem backup.

Feedback Loops: Approve/Reject

O sistema de feedback faz o agente **aprender com suas decisões**. Quando você rejeita uma sugestão, o motivo é anotado — e consultado antes da próxima sugestão.

☐ Como funciona

```
{  
  "date": "2026-02-13",  
  "context": "Sugeri formato longo pra LinkedIn",  
  "decision": "reject",  
  "reason": "Tom muito formal, Bruno prefere casual",  
  "tags": ["linkedin", "tom"]  
}
```

4 domínios: content, tasks, recommendations, digest. Max 30 entradas por arquivo (FIFO). Consolidar padrões em lessons mensalmente.

Granular, JSON

→

Lessons

Curado, prose

→

Decisions

Permanente

Busca Semântica & Otimização

Memory Search — Busca sob demanda

O agente não precisa carregar TUDO na memória. Ele usa busca semântica pra puxar só o que precisa:

Comandos úteis

```
memory_search("decisão sobre modelo pra crons")
→ Retorna ~400 tokens/chunk relevante
```

```
memory_get(path="memory/decisions.md", from=15, lines=10)
→ Puxar trecho específico
```

```
openclaw memory index --all
→ Indexar todos os arquivos de memória
```

Otimização de Tokens

Session initialization rule: carregar APENAS SOUL.md, USER.md, IDENTITY.md e memory/YYYY-MM-DD.md. Usar memory_search() sob demanda pro resto. Reduz de 50KB → 8KB por sessão (~80% economia).

Consolidação Periódica

A cada 15 dias, o agente faz uma revisão completa:

1. Ler todas as notas diárias recentes
2. Identificar lições e decisões que escaparam
3. Atualizar topic files com o que vale manter
4. Atualizar MEMORY.md com aprendizados destilados
5. Deletar notas táticas vencidas (>30 dias)

Pense como um humano revisando seu diário e atualizando seu modelo mental. **Notas diárias são registros brutos. MEMORY.md é sabedoria curada.**

Checkpoint do Módulo

- Pasta `memory/` criada com 5 topic files
- MEMORY.md como índice na raiz
- Notas diárias automáticas configuradas
- Compactação configurada (160k context, 30k reserve)

- Regra de extração antes de compactação no AGENTS.md
- HEARTBEAT.md com checklist periódico
- Feedback loop configurado (pelo menos 1 domínio)

"Se não extrair antes de compactar, perde 80% do valor."

— Lição do Dia 4



PRÓXIMO MÓDULO

Módulo 5 — Integrações & Cronos: Conectando ao Mundo Real