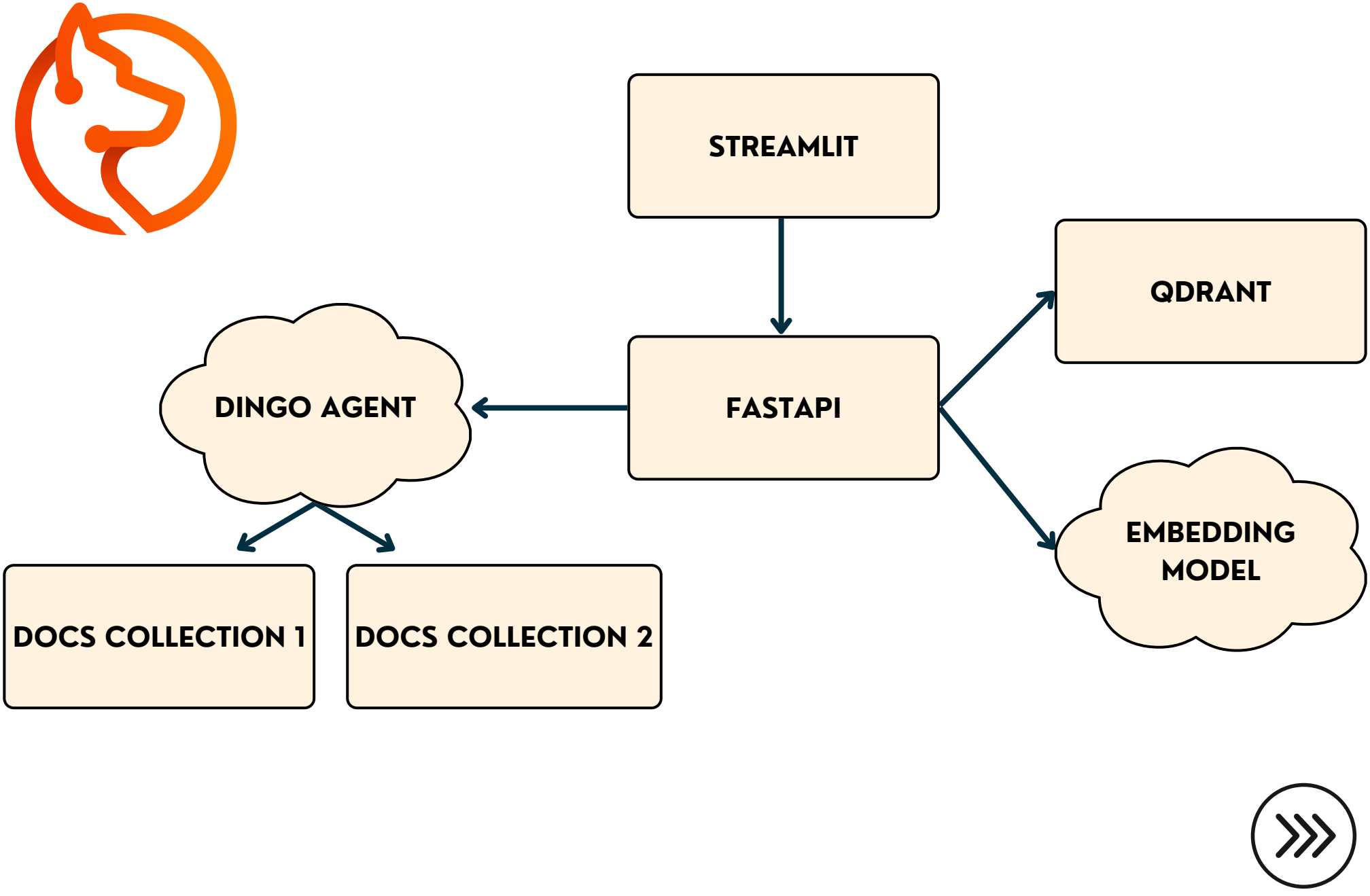
Application Architecture



Indexing

build.py

```python
# Read the content of the external document (e.g., website)
reader = WebpageReader()
docs = reader.read("https://...")

# Chunk the document
chunker = RecursiveChunker()
chunks = chunker.chunk(docs)

# Embed the chunks
embedder = OpenAIEmbedder()
embedder.embed_chunks(chunks)

# Populate vector store with embedded chunks
vector_store = Qdrant()
vector_store.upsert_chunks(chunks)
```

Retrieval and Augmentation

serve.py

```python
agent = Agent(llm, max_function_calls=3)

# Define a function that an agent can call if needed
@agent.function
def retrieve(topic: str, query: str) -> str:
    if topic == "topic_1":
        vs = vector_store_1
    elif topic == "topic_2":
        vs = vector_store_2
    else:
        return "Unknown topic."
    query_embedding = embedder.embed(query)[0]
    retrieved_chunks = vs.retrieve(k=5, query=query_embedding)
    return str([chunk.content for chunk in retrieved_chunks])

# Create a pipeline
pipeline = agent.as_pipeline()
```

🎉 RAG Agent

🦊 Agent

👤 Hello!

🦊 Hello! How can I assist you today?

👤 How many parameters does the Phi-3-mini model from Microsoft have?

🦊 The Phi-3-mini model from Microsoft has 3.8 billion parameters.