Scalable Data Warehouse Pipeline Logical Design

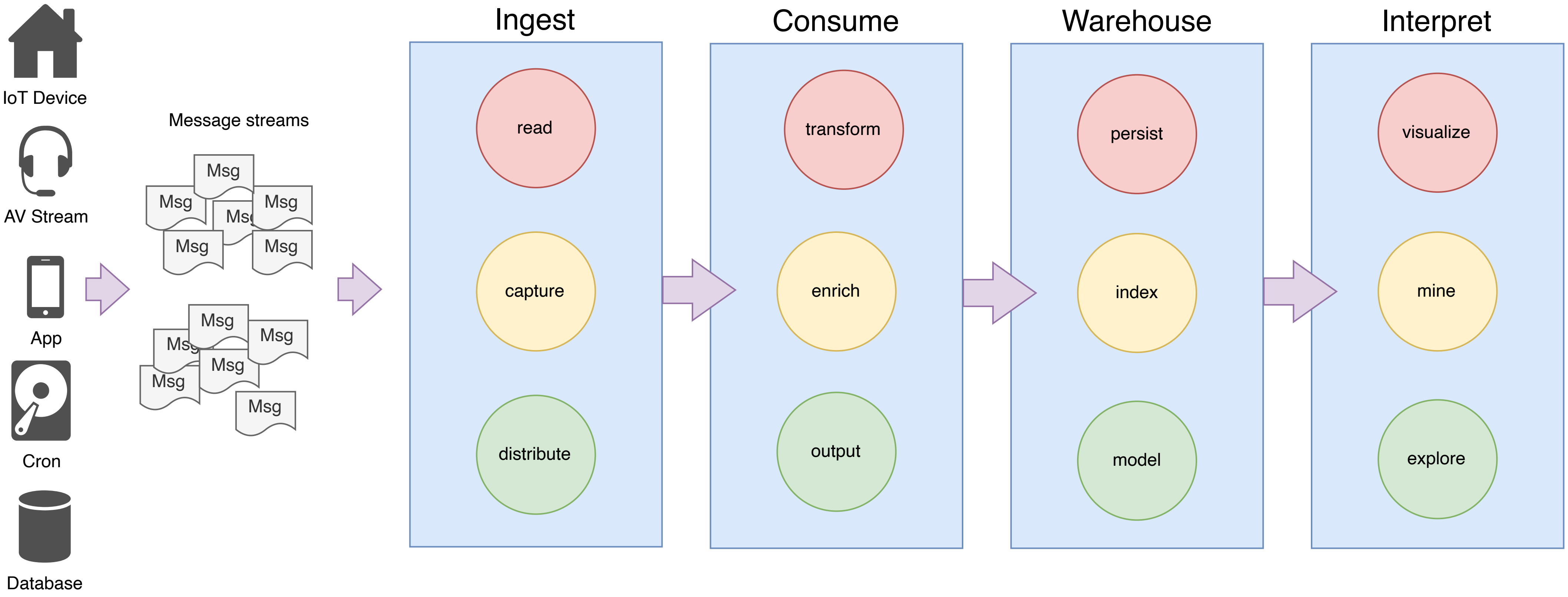
1. The data can be produced by a wide array of sources, but, first, in order to be processed, the data must be transformed into a message. The message represents the contract of the pipeline with the outside world. This contract can be well defined with many different schemas or very loosely defined with only a few primitives required by the message provider.

2. Once created, these messages are then sent for ingestion. Messages can be streamed, batched or both. The ingestion process must not only be able to quickly store the messages, but also must be able to quickly distribute them. Log based systems such as Kafka or Google Cloud Pub/Sub are well suited for this process.

3. Consumers are then notified of or periodically check for new messages. The data in the message is then validated, updated, enriched, transformed and acted upon by the consumers and ultimately outputting the data to the warehouse. Separating ingestion and consumption is key to being able to scale and process cost effectively for large volumes of messages.

4. The warehouse then indexes and stores the data so that it can be efficiently utilized at scale. Security and models to enable self service capabilities should be considered when designing the warehouse to reduce toil and increase business agility.

5. Humans and machines make use of the data warehouse models with reports, analytics, and AI/ML tools. Data mining and further ad hoc analysis also often happens at this stage.


IoT Device
AV Stream
App
Cron
Database
Message streams
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Msg
Ingest
read
capture
distribute
Consume
transform
enrich
output
Warehouse
persist
index
model
Interpret
visualize
mine
explore