

CompanyGraph — Your 📊 landscape, in ArchiMate's words

Systems and their kinds, connections, data and services, written as pages a check holds and mapped to ArchiMate in both directions.

RESTS ON concept Pack

Robert Blust · Software Engineer & Architect

The method you know

The pack writes ArchiMate's elements down as pages, and its README maps them both ways.

THE OPEN GROUP

[ArchiMate Model Exchange File Format](#)

ArchiMate® is a registered trademark of The Open Group. The pack uses the name to say what its types align to; it reproduces no part of the specification, and nothing here is certified by The Open Group.

RESTS ON

concept Pack

A system and its kind

One type where ArchiMate has six elements. Its kind is a page in your company's words that carries the ArchiMate element once.

RESTS ON

system-kind Deployment

system-kind Platform

system-kind SaaS

model/system-kinds/deployment.md

id: 01a1256e-57e2-7acc-a706-3177ef8b8dac

source: Local

rank: 10

element: application-component

Deployment

> Software CompanyGraph builds, releases and deploys to a platform it rents or takes free.

...

A system

A system names its kind, the seat that owns it, the data processor behind it and what it runs on.

RESTS ON

system MCP server

system Cloud Run

model/systems/mcp-server.md

```
---
id: 01a1256e-8662-7925-9097-a884e87a2658
source: Local
kind: Deployment
owner: Owner
processor: Google Cloud
part-of: Cloud Run
---

# MCP server
> Answers questions about CompanyGraph's own model, read at one pinned commit when its
  image is built and not live, over the Model Context Protocol at mcp.companygraph.io, for
  any agent that connects and for the chat.

...
```

What it connects to

A row is written on the side that takes the data, with the interface's name in **As** and the service it calls in **Service**.

RESTS ON

system Chat server

service MCP endpoint

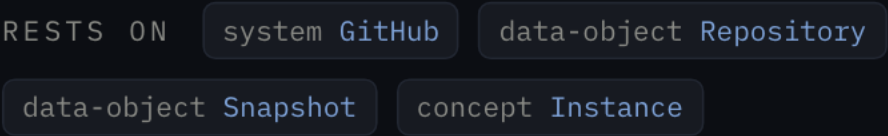
system Claude API

```
model/systems/chat-server.md

...
### Connects to
| System | As | Service | Carries | Via |
| --- | --- | --- | --- | --- |
| MCP server | Streamable HTTP | MCP endpoint | Entity | HTTPS |
| Claude API | Messages API | | | HTTPS |
| TypeSafe | System One API | | | HTTPS |
...
```

What it holds

A row names a concept, the data object that keeps it and the access. One system holds each concept as `master`.



```
model/systems/github.md

...
# GitHub
> Holds CompanyGraph's repositories, among them the vocabulary, the tools and its own
  model, with their pull requests, issues and releases, for the seats that work on them.

### Holds
| Concept | Data object | Access |
| --- | --- | --- |
| Instance | Repository | master |
...
```

A service

What a system exposes to others, provided by the systems that expose it, realizing the feature the business sees.

RESTS ON

service MCP endpoint

feature An agent reads the model

model/services/mcp-endpoint.md

id: 01a1256f-009b-7dca-b327-76f8b78a27cf

source: Local

provided-by:

- MCP server

realizes:

- An agent reads the model

MCP endpoint

> The Model Context Protocol endpoint at mcp.companygraph.io/mcp, where any agent client, and the chat, lists the types CompanyGraph's model declares, reads its entities and searches it, without signing in.

...

The mapping

A generator reads it both ways: from a tool’s export into pages, and from pages back into the exchange format.

RESTS ON concept Pack

Application Component, Node, System Software, Device, Equipment, Communication Network	a system, whose kind’s element names which one it is
Application Interface, Technology Interface	the As of a Connects to row
Data Object	a data-object, realizing its business object, a core concept
Access, with its mode	a Holds row: writes or reads, the leading writer master
Function, Artifact	dropped: internal behavior nobody can name a user of

Where it departs

Where the pack does not do what ArchiMate does, its README says so and says why.

RESTS ON concept Pack

Six elements	one type, the kind's <code>element</code> naming which one, so the element comes back exact
An interface	a row, not an element; its name survives in <code>As</code>
Triggering	comes back as Flow: a row says what is carried and how, not whether it starts something
Functions and artifacts	no page, by core's rule that a thing nobody can name a user of is not a feature
Access	<code>readwrite</code> is <code>writes</code> , since a writer reads, and <code>master</code> is a claim ArchiMate does not make
Lifecycle	Four stages, <code>planned</code> , <code>active</code> , <code>retiring</code> , <code>retired</code> : the model says what is, not when it will be

How the pack joins core

A system serves a process or realizes a feature, a service realizes a feature and a data object realizes a concept. A system names the seat that owns it and its data processor. Every edge runs one way and is optional.

RESTS ON

process	Answering	feature	An agent reads the model
concept	Instance	seat	Owner
		data-processor	Google Cloud

LEVEL 1 · LANDSCAPE **system · service · data object**

↓ serves · realizes · owner · processor ↓

LEVEL 0 **Core**
process · feature · concept · seat · data processor

Left for later

What the pack does not have yet is named in its README, not hidden.

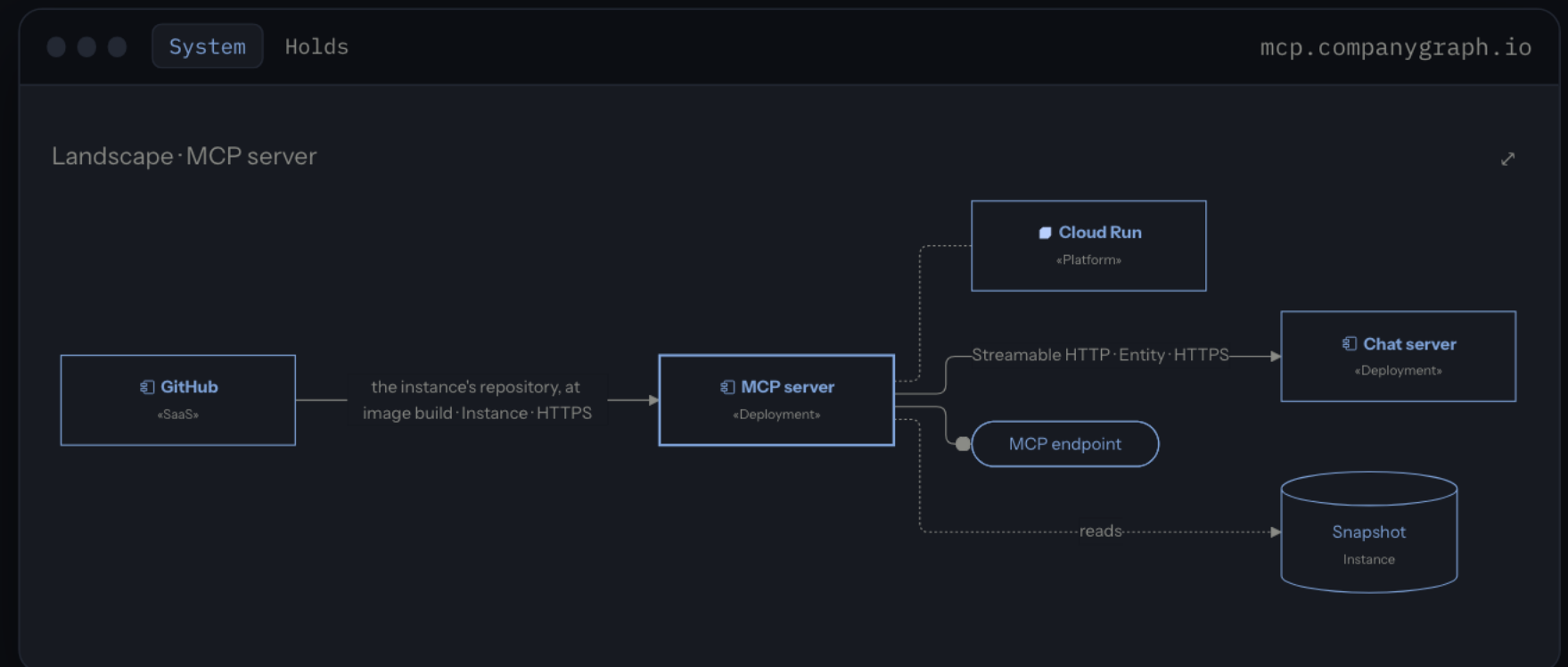
LATER	Location, as a type of this pack
LATER	an integration type
LATER	functions
LATER	license and cost
LATER	a collaboration, ArchiMate 4's element for several systems working together
LATER	a notation on the kind
LATER	an artifact realizing a data object
LATER	a field list on a data object
LATER	a generator from an architecture tool's export, which is an instance's own

RESTS ON concept Pack

The landscape, drawn

Nobody drew these. The MCP server's own code draws them from our pages at the commit this site pins — the same code that answers an agent at mcp.companygraph.io.

RESTS ON `system MCP server` `system GitHub` `data-object Snapshot`
`data-object Repository`



Your landscape, as pages

ArchiMate's elements as pages a check holds, mapped in both directions. Where the pack departs, its README says why.

RESTS ON

concept Pack

LEVEL 1 · LANDSCAPE **system · service · data object**

↓ serves · realizes · owner · processor ↓

LEVEL 0 **Core**

process · feature · concept · seat · data processor