Cloud Concepts for Unix Gurus

Joe Epplin, Cloud Engineering Practice Lead, OCI

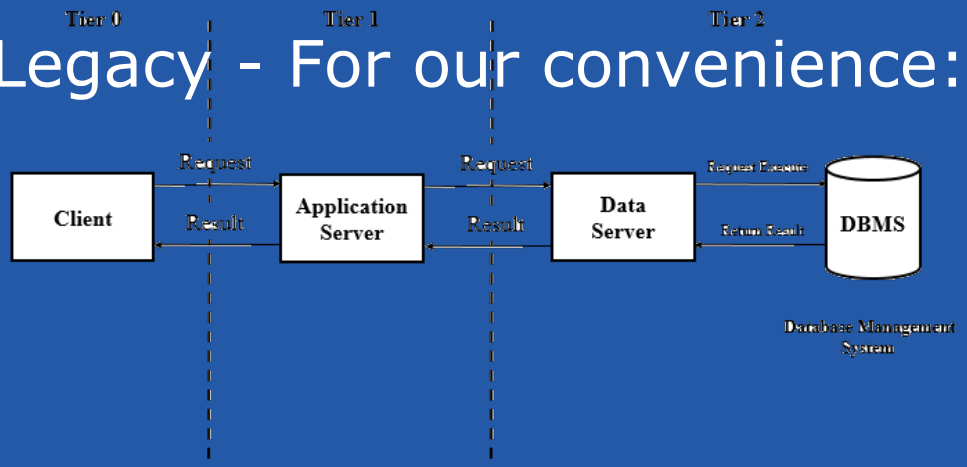
Agenda

1. Core Cloud Concepts
2. How Is This Better?
3. But What About Security?
4. Differentiating Cloud Vendors
5. Why All The Buzz About Microservices?

Core Cloud Concepts

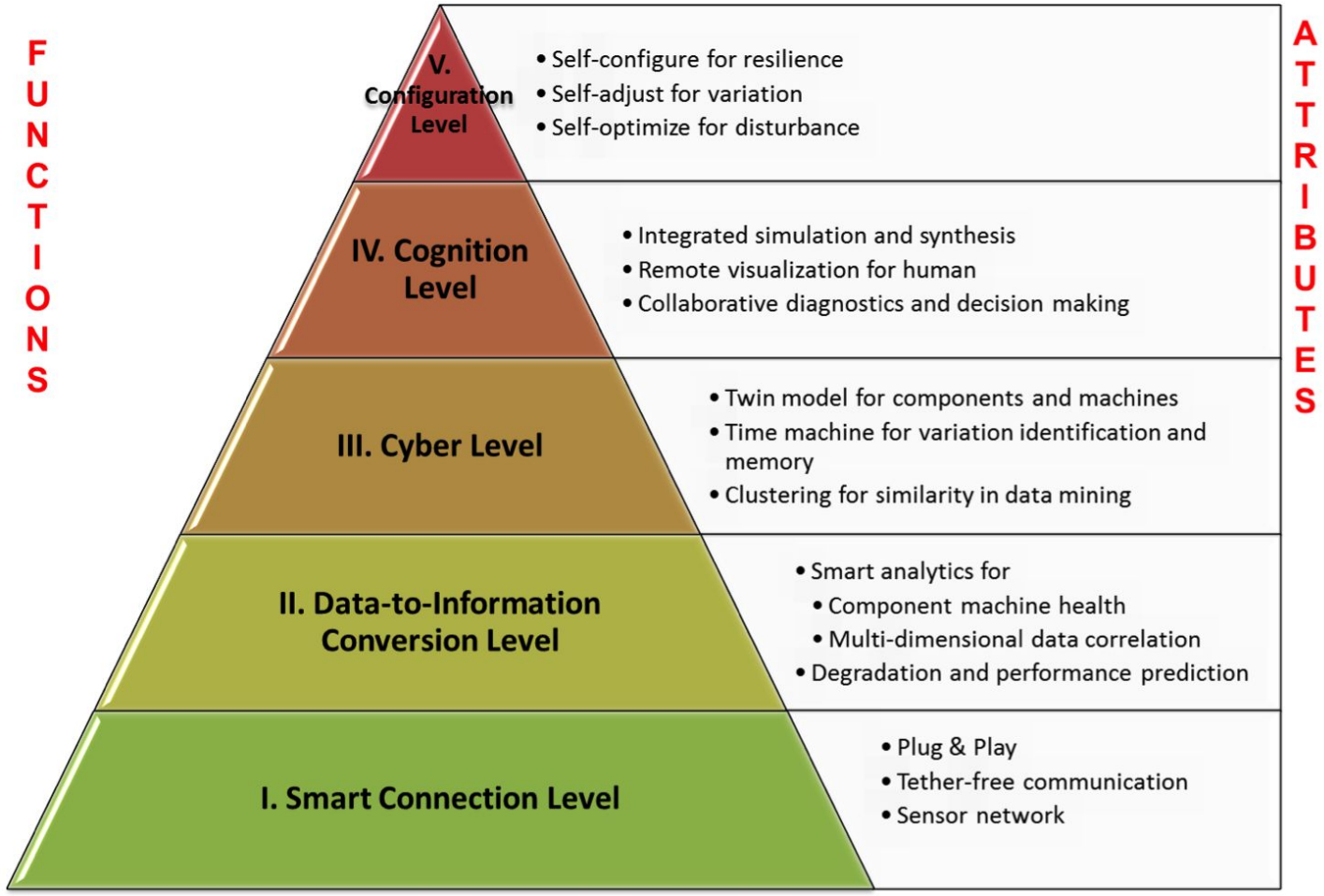
OBJECT COMPUTING

Legacy - For our convenience:

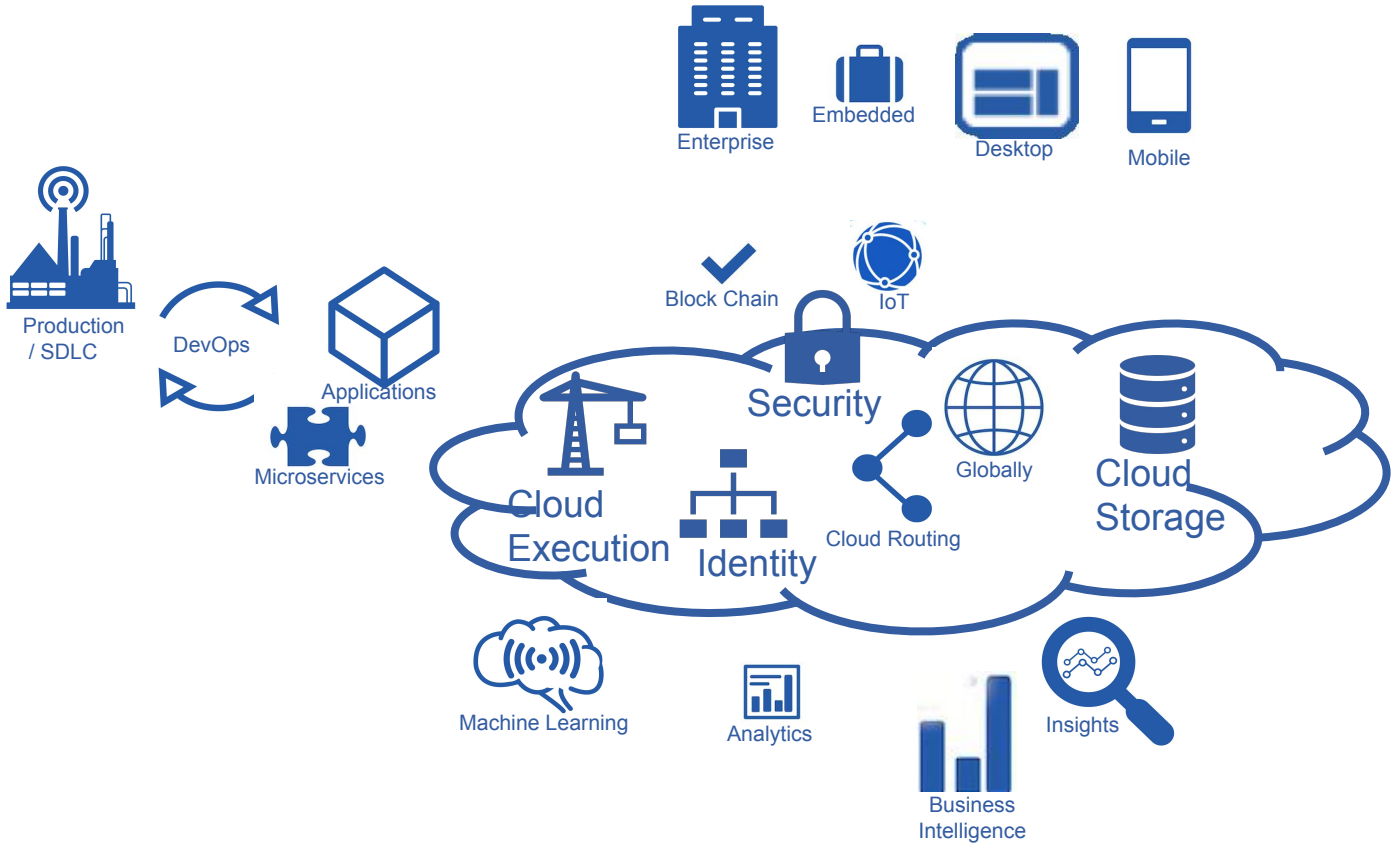


Architect Like You Mean It

How we need it to be:



Everything is a Service Component



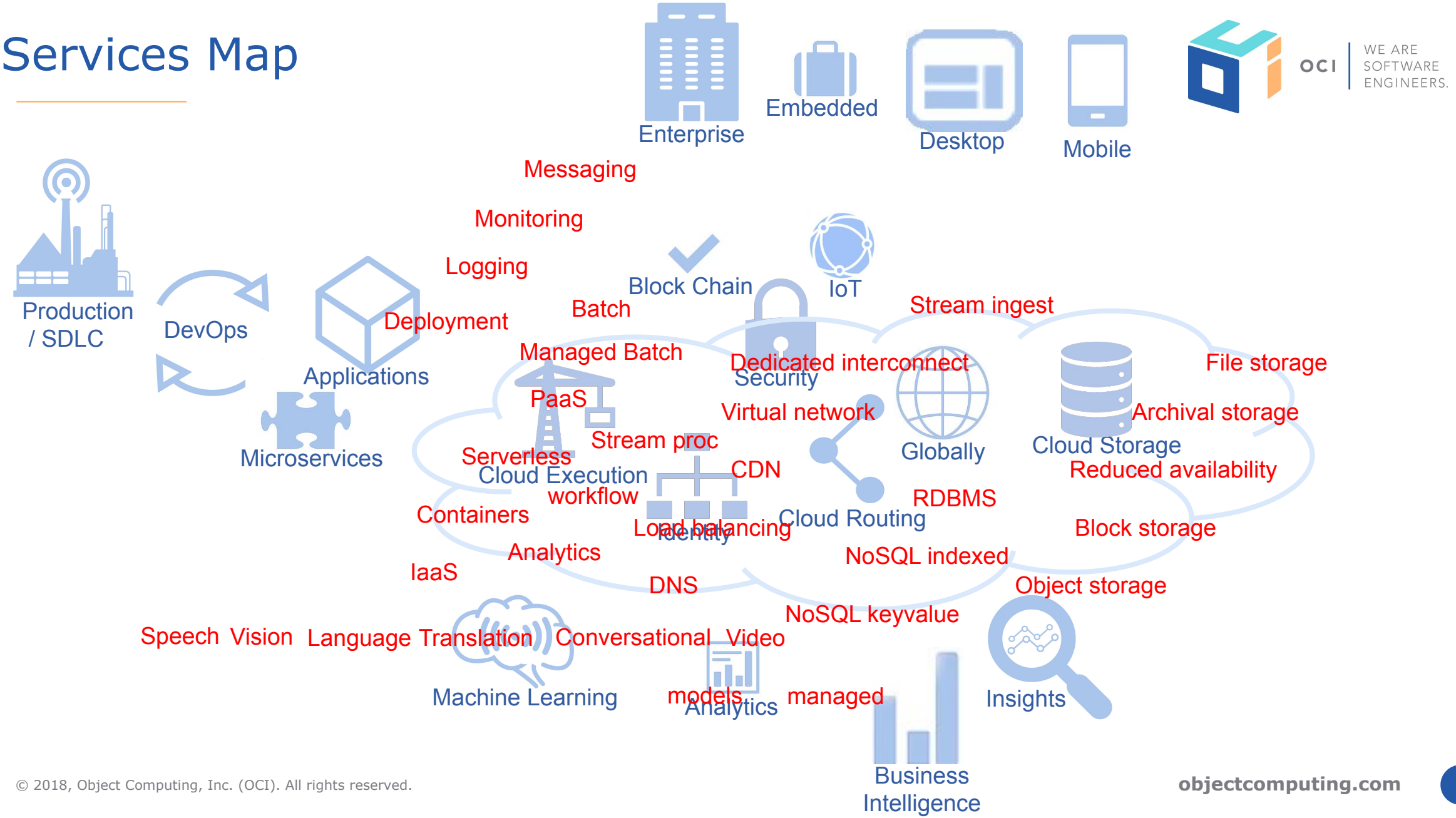
Yes. Everything!

The #1 idea is that what we do as system admins can in itself be services, if we think of these as components.

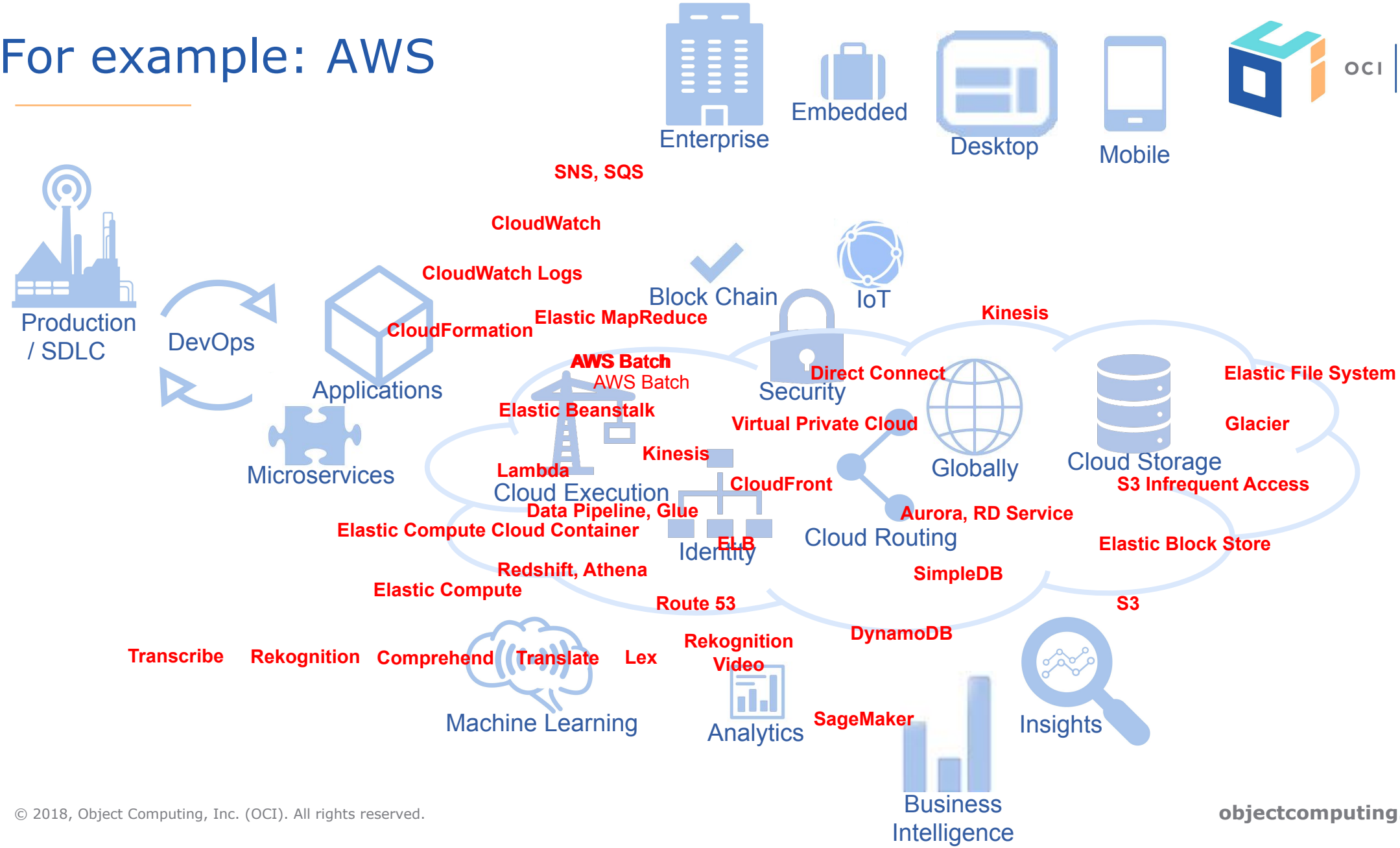
Architect the interactions we have to support systems into the interfaces of the cloud services.

The "cloud" abstracts infrastructure complexities of servers, applications, data, and heterogeneous platforms.

Services Map



For example: AWS



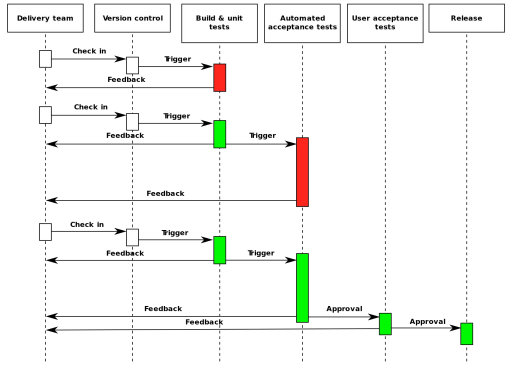
How Is This Better?

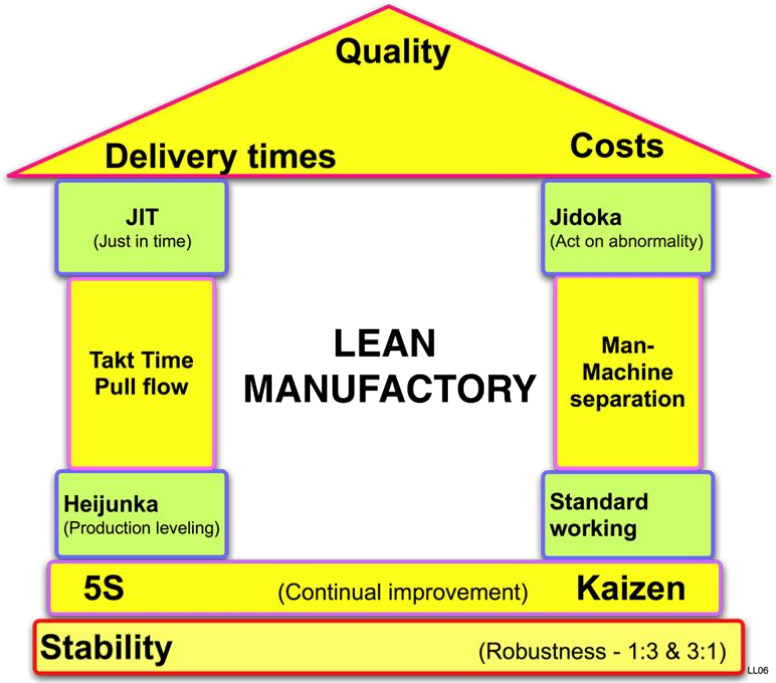
OBJECT COMPUTING

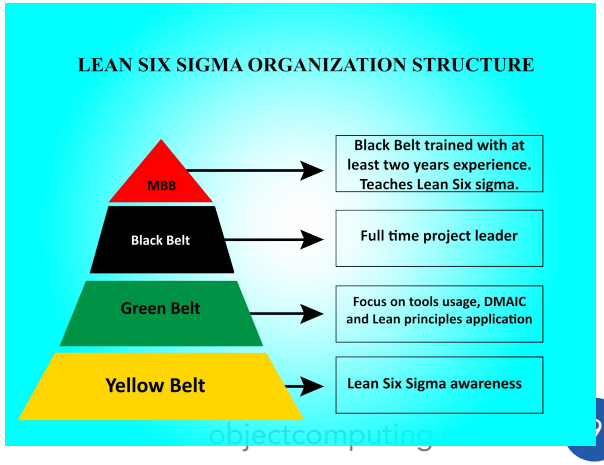
JIT Foundations

Acquisition of the required hardware is removed from the "time-to-market".

- Try out a new OS - no problem!
- Stand up a new JVM - what else do you need?
- Support an experimental product - Let's go get breakfast.







Levels of Control

- **IaaS** (Infrastructure as a Service): Virtual hardware is provided and managed by the cloud provider.

  Most system administrative roles still performed by web services client teams

- **PaaS** (Platform as a Service): Both hardware and an operating system are provided and managed by the cloud provider.

  Account management, connectivity, secrets management, still performed by web services client teams

- **SaaS** (Software as a Service): Applications are provided and managed by the cloud provider.

  No system administrative roles still performed by web services client teams

When we talk about cloud service models, we're referring to a virtual infrastructure where things like software, applications and storage are hosted, providing varying levels of access to individuals or businesses.

OBJECT COMPUTING

But What About Security?

OBJECT COMPUTING

"Only The Paranoid Survive"

It is under your control, AND you have to control it.

New Security Paradigm

New roles:

- Cloud provider
- Cloud user
- Data owner

Similarities and differences to traditional outsourcing

Some key issues:

- Trust
- Multi-tenancy
- Encryption
- Compliance

Advantages

1. Data Fragmentation and Dispersal
2. Greater Investment in Security Infrastructure
3. Fault Tolerance and Reliability, Greater Resiliency
4. Simplification of Compliance Analysis
5. Data Held by Unbiased Party (cloud vendor assertion)
6. Low-Cost Disaster Recovery and Data Storage Solutions
7. On-Demand Security Controls
8. Real-Time Detection of System Tampering
9. Rapid Re-Constitution of Services
10. Cross-enterprise coordinated bad actor identification

OBJECT COMPUTING

Better security is affordable

OBJECT COMPUTING

Seem reasonable

Challenges

Truly caused by Cloud

1. Need for isolation management
2. Attraction to hackers (high value target)
3. Possibility for massive outages

Side-effect of Cloud Benefits

1. Multi-tenancy
2. Logging challenges
3. Data ownership issues
4. Quality of service guarantees
5. Encryption needs for cloud computing
   - Encrypting access to the cloud resource control interface
   - Encrypting administrative access to OS instances
   - Encrypting access to applications
   - Encrypting application data at rest
6. Public cloud vs internal cloud security
7. Lack of public SaaS version control

Differentiating Cloud Vendors

OBJECT COMPUTING

Myths That Need To Die

1. This is too complicated

- Existing solutions do not correctly represent their requirements; they were the best at that time

2. Cloud services are primarily about hosting

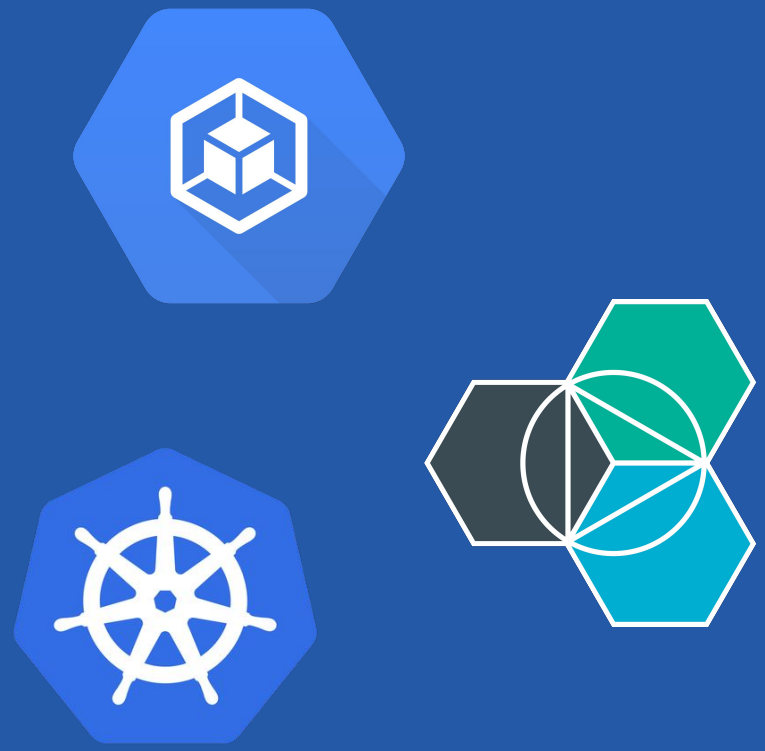
- 17% of current cloud services perform hosting
- 24% of those offer more value than hosting

3. This is spooky

- Competition across vendors has driven simplicity, clarity, and maturity
- Having a guide with broad knowledge and deep experience is much more efficient

So, Who's Good For What?

- All six top vendors are viable
- Each "Top-3" vendor has unique positioning
- De facto standards for essential cloud services have emerged; enterprise consumers can expect comparable behavior for storage, database, infrastructure, platform, containerization, kubernetes, identity management, routing, security, and machine learning.
- So ... viewing the cloud services vendors like vendors has major advantages for enterprises; purchasing understands how to arrange the best terms while comparing costs and benefits of the providers



So, Who's Good For What?

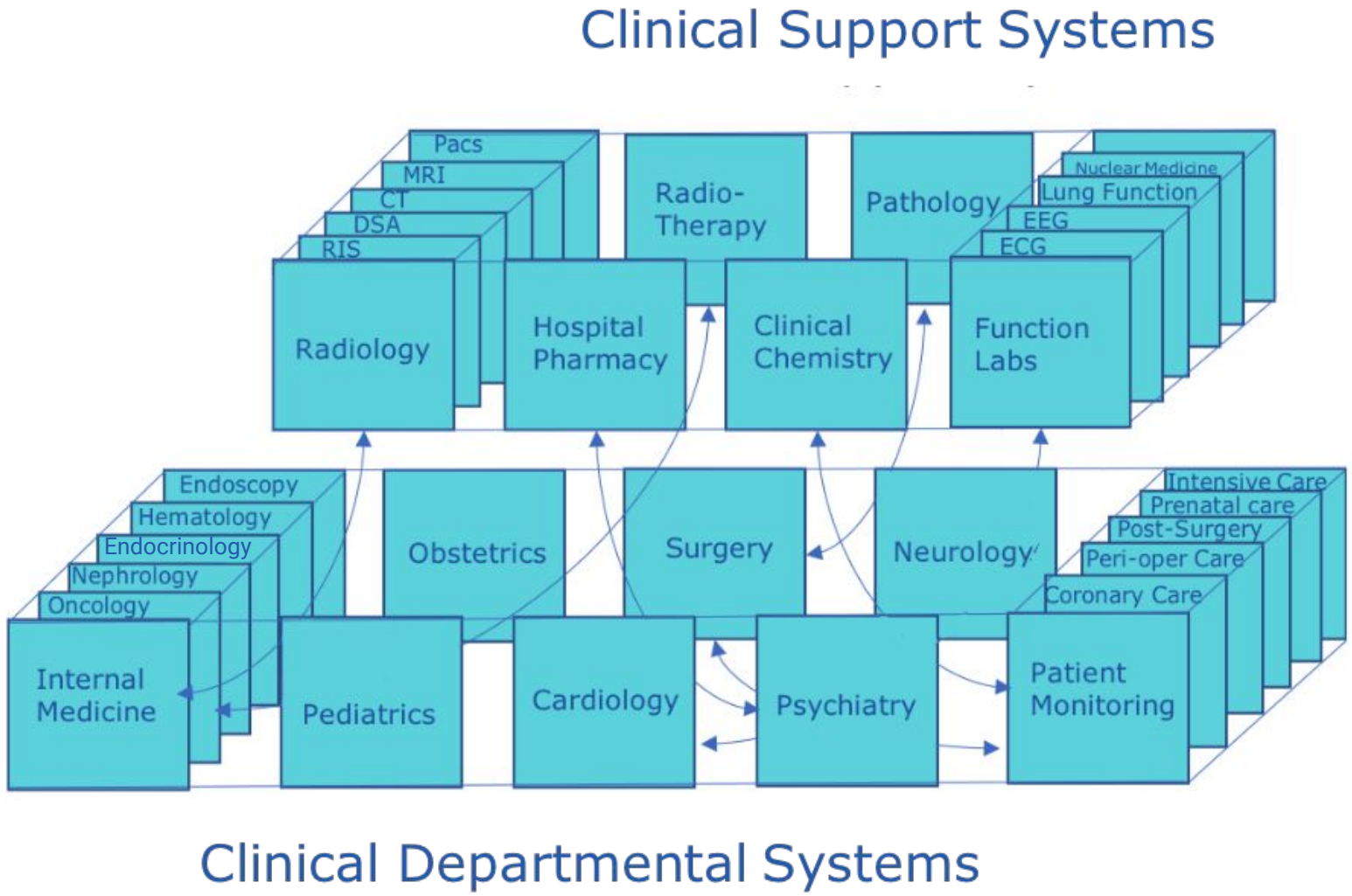
- Unique positioning
  - AWS - all purpose, heavy lifting, all enterprise
  - Azure - all business including medium to small business,
    common sense utility, 4-12% cheaper
  - Google - Global reach of information: geographic, language, weather, research, demographic, visual, taxonomy, unified administration
  - The rest - specialty niches

OBJECT COMPUTING

Why All The Buzz About Microservices?

Monolithic Legacy Systems

Example Monolithic System

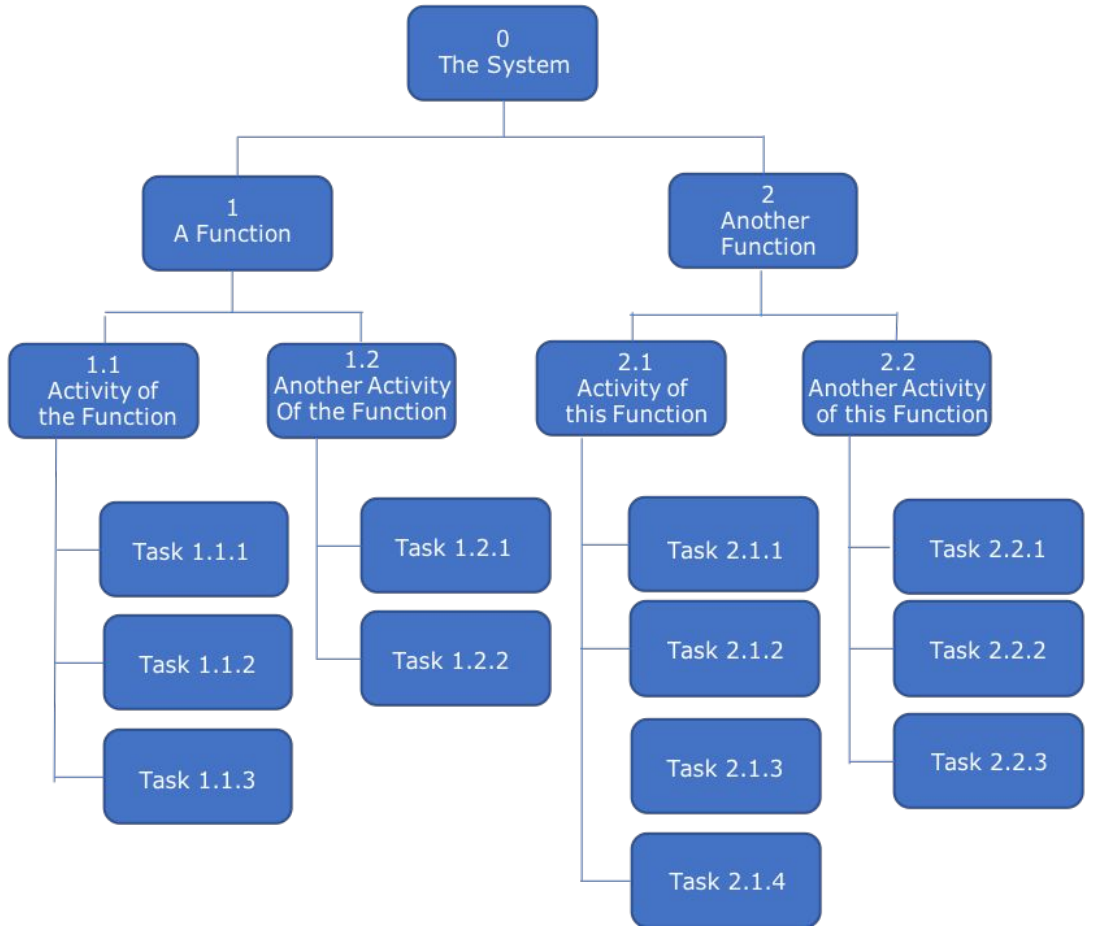


These Systems Support Business Processes

OBJECT COMPUTING

There is a taxonomy for these processes.

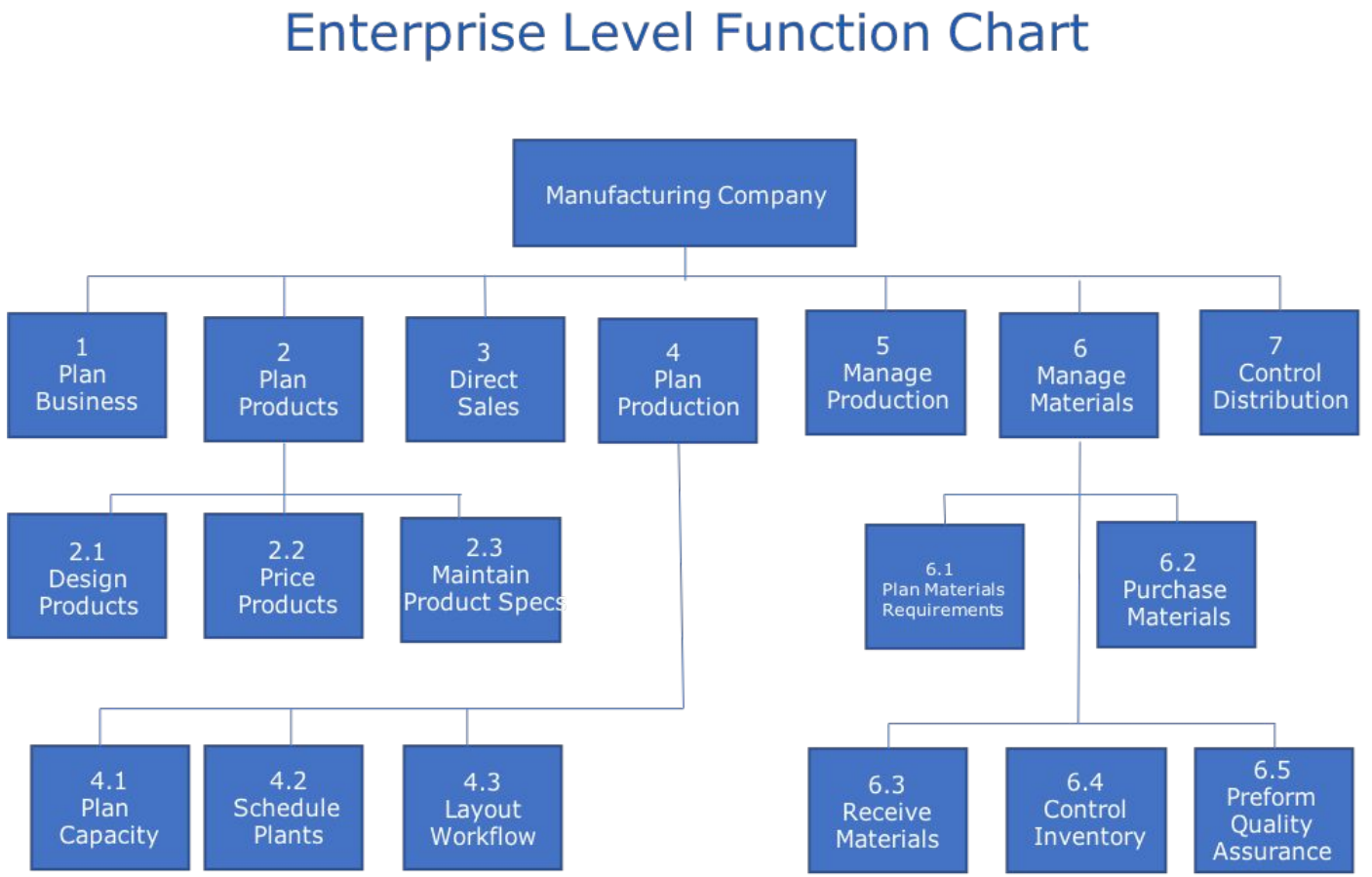
Decomposition Diagram (Hierarchy Chart)



Sample Decomposition

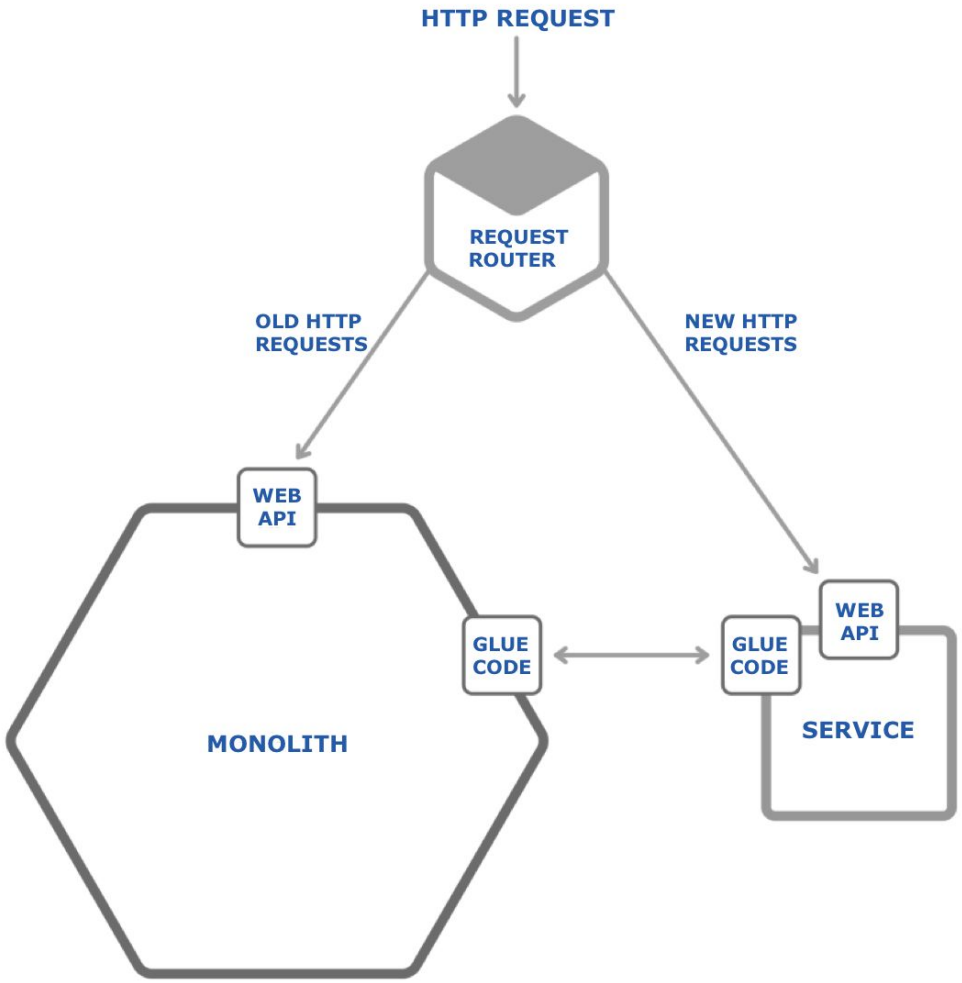
OBJECT COMPUTING

There is a taxonomy for these processes



Note Conway's Law

Monoliths and Microservices Can Coexist



Strangler pattern

An Example Framework that facilitates microservices

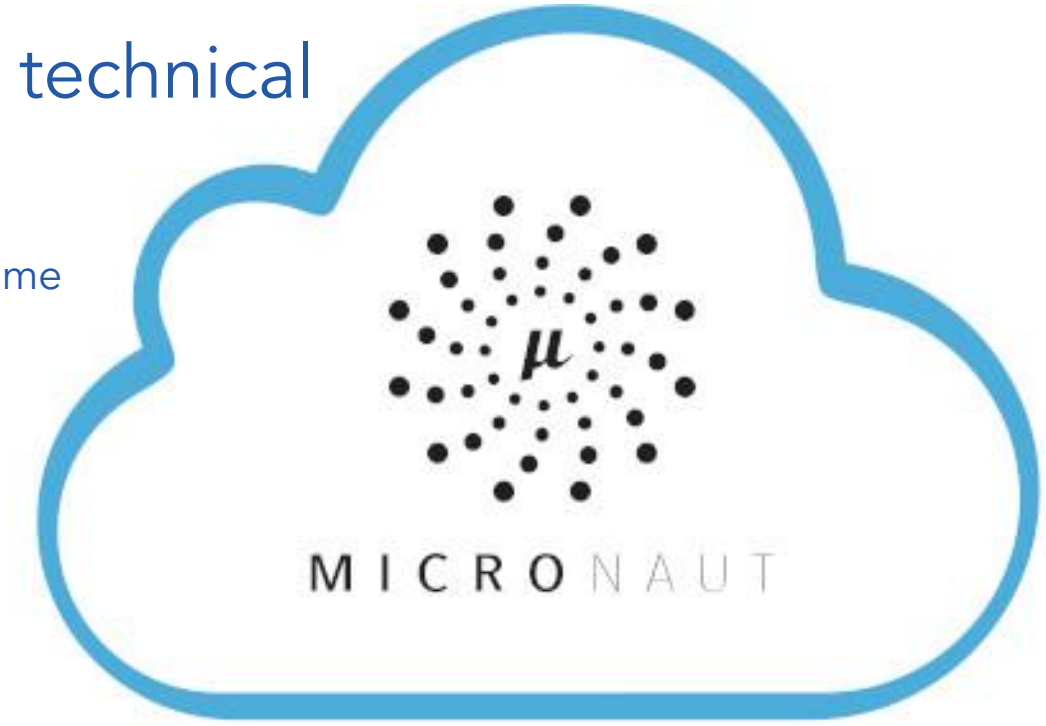
OBJECT COMPUTING

By addressing the technical aspects

Declarative, reactive, compile-time HTTP client

Non-blocking HTTP server built on Netty

Fast and easy testing



Efficient compile-time dependency injection and AOP (not reflection)

Fully reactive and non-blocking apps

Built in cloud support (common discovery services, distributed tracing tools, and cloud runtimes)

A Recipe For Microservices Adoption

OBJECT COMPUTING

Business Process Decomposition / Refactoring

1. Model systems with highest business value first
2. Do something concrete with the model as quickly as possible
3. Grow from there
4. Highlighting business process inefficiencies will inform the microservice architecture
5. Diagrams should fit on one page
6. Manage different versions of the diagram for different stakeholders
7. BPD depicts information flows in business processes, not data flow; let the business organization vision inform the structure of the emerging services

Where Can I Get Help?

End