Objectives

#1 look at: current API-centric

#2 introduce: Cell-based architecture
Motivation
Reality of the Enterprise
Brownfield > Greenfield

Legacy, monolithic → Microservices, sprawl

Reference Implementations

picture credit: https://www.marinersmuseum.org/
Underutilization of the Technology
Gap: architecture | development | deployment
Dependency management
Architecture Patterns
Timeline

1970
Mono
UX
BL
Data

1980
2-Tier
UX
BL
Data

1990
3-Tier
UX
BL
C
Data

2000
MVC
UX
BL
Services and APIs
Data

2012
SOA
EDA
WOA
MSA
BL
Data
Background: Layered Architecture
A platform with an agile team
100 APIs, 60 message flows, 80 services, n DBs
Multi-tenanted, 3 active tenants
First release after 3 years
Rise of Microservices
Pragmatic Microservices

- Netflix: APIs
- Uber: Edge Gateway
- eBay: API Facade
- Gartner: Mini Services
Background: Layered Architecture with MSA
Background: Segmented Architecture

Business Communities
- Quality Assurance
- Marketing
- Finance
- HR
- Help desk
- Information management
- Sales

Management Layer
- G-1: Gateway
- G-2: Gateway
- API Life-cycle
- Monitoring
- Catalog

Presentation Services
- Business Domain Services
- Utility Services
- Storage
- RDBMS
- NoSQL
- Files
- in-memory

Governance
- Behaviours
- Controls
- Certification
- Policies
- Enforcement
Platform of Platforms

CI/CD
User Store

Platform (shared capabilities)
Project 1  Project 2  Project 3  Project n

Platform (shared capabilities)
Project 1  Project 2  Project 3  Project n

Platform (shared capabilities)
Project 1  Project 2  Project 3  Project n

Platform (shared capabilities)
Project 1  Project 2  Project 3  Project n
Making of......

picture credit: https://commons.wikimedia.org/wiki/User:Frank_Schulenburg CC BY 3.0
Building the Concept

_quantum computing_kubernetes_

[bacteria image]

[DNA image]
Business vs technical services
Service: Technical definition

A **code** exposes through an **interface** that describes a collection of operations that are **network accessible** using a standardized messaging protocol.

```java
public class PersonApi {
    @Path("/greet")
    @GET
    @Consumes(MediaType.APPLICATION_JSON)
    @Produces(MediaType.APPLICATION_JSON)
    public JSONObject sayHello() {
        try {
            return new JSONObject().put("greeting", "Hello world");
        } catch (JSONException e) {
            return null;
        }
    }
}
```
Service: Business definition

Software components that can be spontaneously discovered, combined, and recombined to provide a solution to a business problem.
Microservice: Technical definition

A microservice must have a **single purpose** and be loosely coupled in design and deployed independently of other microservices.

“Micro” is a concept of **scope** rather than **size**.
Microservice: Business definition

Microservices is an approach to application development in which a large application is built as a suite of modular components or services. These services are built around business capabilities.
Group of (Micro)services
The cell is the basic structural, functional, and biological unit of all known living organisms.
Cell-based Reference Architecture
Component: Atomic Units

A component represents a process or business logic running in a container, serverless environment, or an existing runtime. A component is designed based on a specific scope, which can be independently run and reused at the runtime.
Cell: Units of Enterprise Architecture

A **cell** is a collection of components, grouped from design and implementation into deployment. A cell is independently deployable, manageable, and observable.
Cell:Component
1:M
1:1
Connected Cells

picture credit: https://www.medicalnewstoday.com/
Control Plane:
- Signaling of the network
- Makes decisions about the traffic flow

Management Plane:
- Configure
- Observeability, Monitor

Data Plane:
- Forwards traffic between hops
- Takes data packets

picture credit: https://www.flickr.com/photos/teflon/
Inter and Intra Cell communication

Global-mesh

Cell-mesh
Connected Cells

Cell gateway (ingress)

Sidecar (egress)

Adaptor (egress)

Ambassador (egress)
API-centric Architecture

Pull APIs
- RESTful HTTP, gRPC

Push APIs
- Events JMS, AMQP, SMTP
- Streams Kafka, MQTT
Gateway Pattern
Automated governance is made of three things:

A source of truth:
  Policy store/registry

Enforcement of the policy
  Gateway or plugin attempting to keep the desired state

Observability
  How close to the desired state are we?
Security of Cells

Security of Cells
Developer Experience (DX) of a Cell
Creating Cells

Brand new Cell

Existing (micro)services

Update an existing Cell

Create a new version
Lifecycle of a Cell
Structured Agility

Versioned Components

Versioned Cells

Dependency managed

Autowired

Reusable

Enhanced MSA & CNA
Cell-based Enterprise Architecture
# Cell Types

<table>
<thead>
<tr>
<th>Cell Type</th>
<th>Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic</td>
<td>Microservices, Functions, MicroGateways, lightweight storages</td>
</tr>
<tr>
<td>Integration</td>
<td>MicroESB or other integration microservices, lightweight storage and/or cache</td>
</tr>
<tr>
<td>Legacy</td>
<td>Existing systems, legacy services</td>
</tr>
<tr>
<td>External</td>
<td>SaaS and partner systems</td>
</tr>
<tr>
<td>Data</td>
<td>RDBMS, NoSQL, File, Message Broker*</td>
</tr>
<tr>
<td>Identity</td>
<td>IDP, user stores</td>
</tr>
<tr>
<td>Channel</td>
<td>Web Apps, IoT, mobile apps</td>
</tr>
</tbody>
</table>
Reference Implementation L0
Reference Implementation L1
Cells and Podular Organizations
Summary: Cell-based Architecture

- Self-contained
- Deployable as a unit
- Independently elastic
- Data, control & management plane
Just a (steady) start

https://github.com/wso2/reference-architecture
Reference Methodology

https://github.com/wso2/reference-methodology

picture credit: https://www.flickr.com/photos/vasile23/
“Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations.”

- M. Conway
THANK YOU

@asankama

https://www.linkedin.com/in/asankaabeysinghe/