vincit

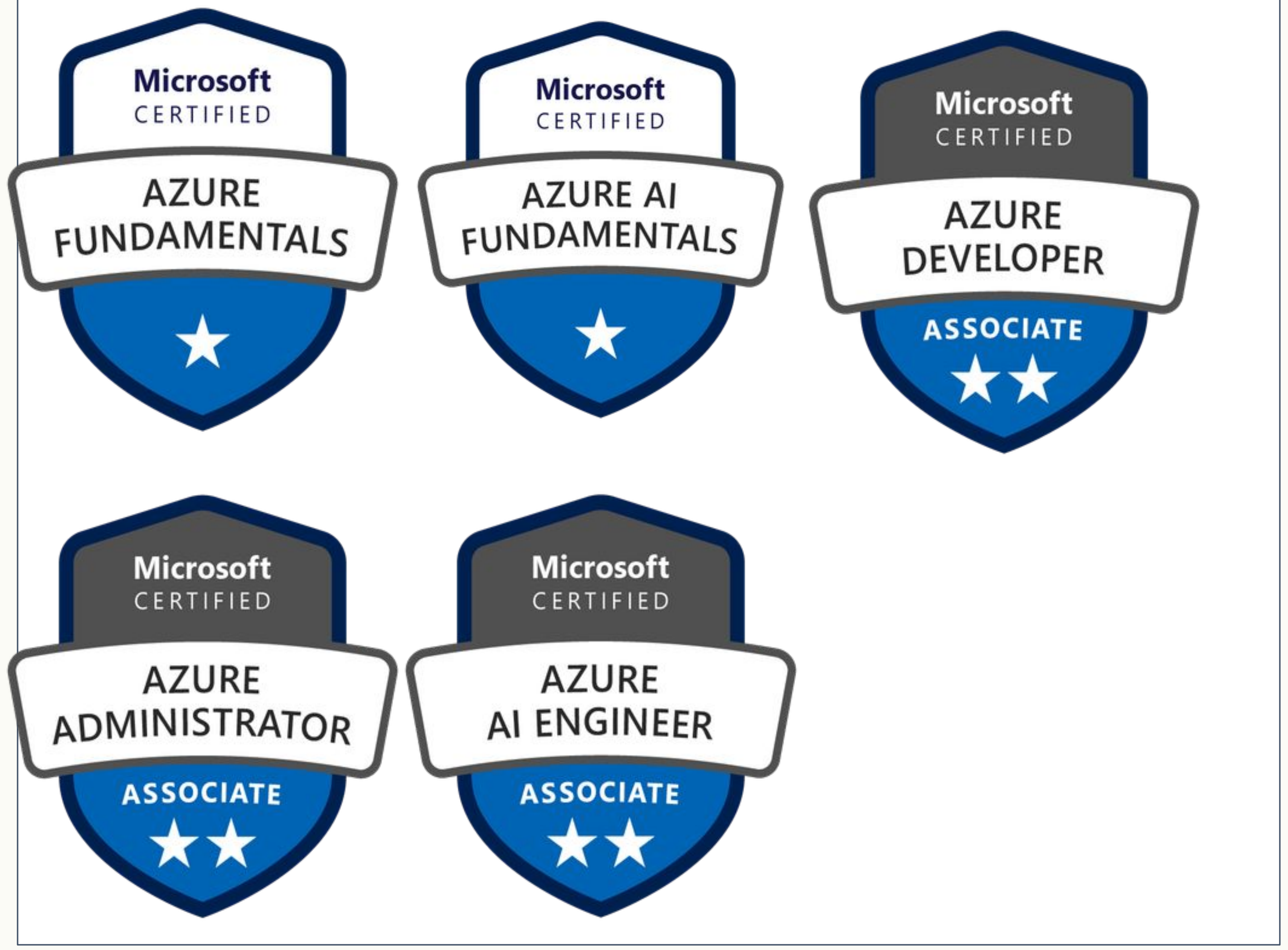
Azure & Friends - case presentation:

Azure solution architecture for global customer in B2B business

Case description from years 2019-2023

TIMO HOLM

Completed Azure-certifications:



Goal 2025

- 🤸 15 years in software industry
- 🤝 7 years in Vincit
- 🦾 Tech lead / architect
- 🚀 5 years of Azure experience in customer projects

CASE: global customer in B2B business 🇪🇺 🇺🇸

📈

Industrial customer business **was growing while selling & licencing their "goods and services"** to their business partners, but their systems were **not scaling for global growth**.

Business partners were giving orders via emails, phone calls and legacy extranet.

👉 Customer wanted a modern, scalable, resilient, secure online system in cloud with excellent end-user experience & capabilities for further development

👉 Customer had also a vision to create API:s for their business partners

🤝 Customer already had 2 long-term IT partners for their ERP and certain business core system

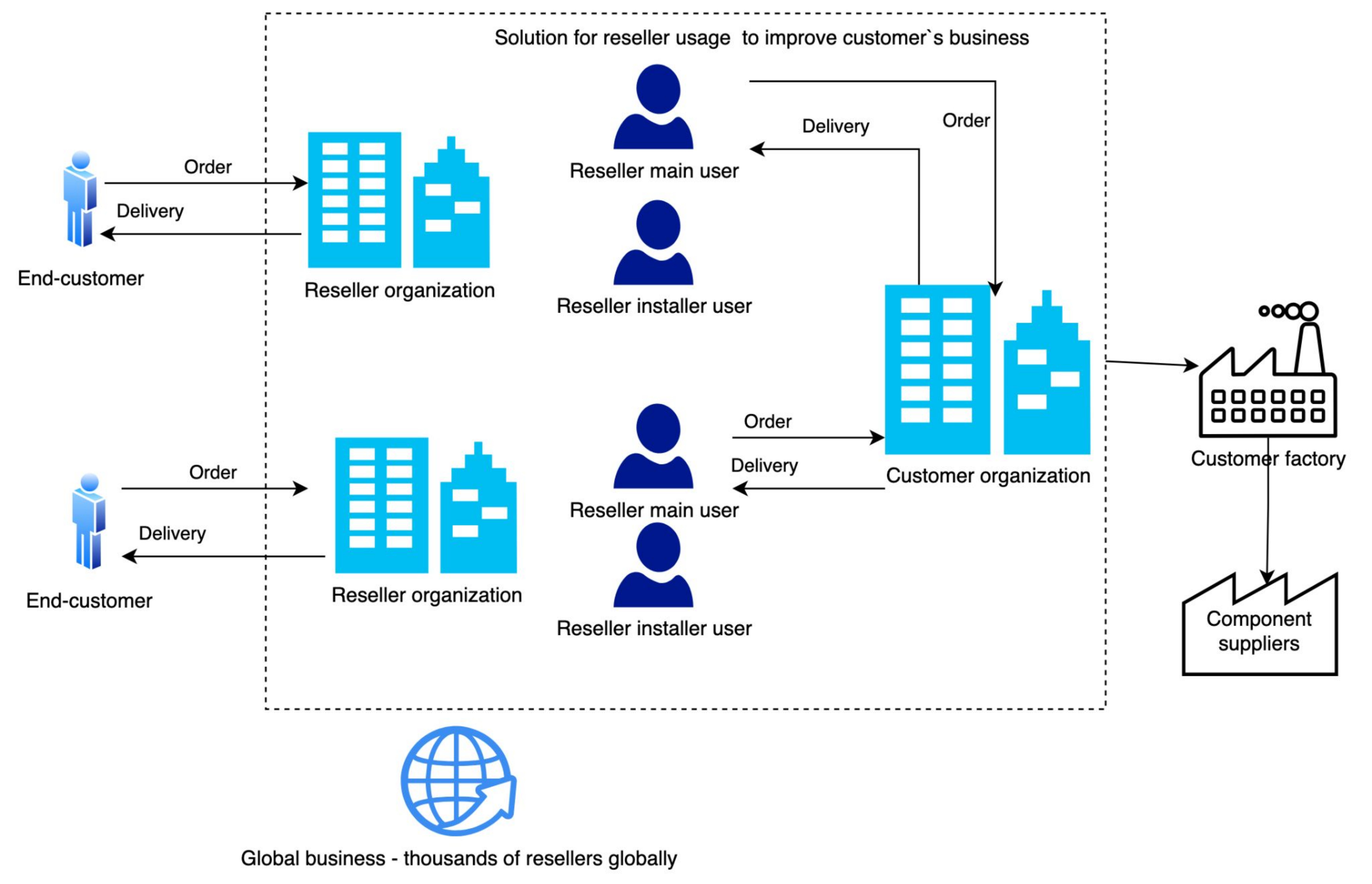
🤷 Customer did not have in-house competence on digitalization of their services.

💰 Customer business size: over 100M eur revenuer per year by selling their "goods and services"

💎 Lots of projectized business needs

🎯 Customer already used Microsoft systems (O365, Azure). Although, their R & D used AWS.

Domain


Solution for reseller usage to improve customer`s business
Order
Delivery
End-customer
Reseller organization
Reseller main user
Reseller installer user
Delivery
Order
Customer organization
Customer factory
Component suppliers
Order
Delivery
End-customer
Reseller organization
Order
Delivery
Reseller main user
Reseller installer user
Global business - thousands of resellers globally

Quality requirements

- “High security”
- Easy access management for various types of users
- Scalability for global growth
- Optimal operational cost structure scaling with growth
- Longevity of the solution & time-to-market
  - Focus on evolving and adapting changes over time
  - Build and release new features on monthly basis
  - Lifetime over 10 years

Solution:

Microservice architecture + Single Page Application in Azure

Infrastructure -as-code via Terraform

🤖 Azure Entra (former AzureAD) B2B

- Federated single sign-on

🤖 Frontend: Single Page Application (SPA)

- Start with Vue migrating to React, NextJS

🤖 Backend: Microservice architecture in AKS

- NodeJS
- Azure Kubernes Service (AKS) running microservices

🤖 Azure API management for API:s

🤖 Azure EventHubs

- for event Pub/Sub via Kafka protocol

🤖 Azure CosmosDb via MongoDb interface

🤖 Azure KeyVault for secrets

🤖 Azure Blob storage for static resources

Azure Kubernetes Service

- Kubernetes, is an *open source system for automating deployment, scaling, and management of containerized applications* [1]
- Azure provides Kubernetes as service, providing features: [2]
  - *Simplified Kubernetes experience*
  - *Curated code-to-cloud experience*
  - *Integrated monitoring and logging*
  - *Advanced security and governance controls*
  - *Cloud to edge deployments*
  - *Secure container supply chains*
- Good pair with microservices [3]:
  - *“Simplify the deployment and management of microservices-based applications with streamlined horizontal scaling, self-healing, load balancing, and secret management.”*

1. https://kubernetes.io/
2. https://azure.microsoft.com/en-us/products/kubernetes-service
3. https://learn.microsoft.com/en-us/azure/aks/what-is-aks

Solution: Microservice architecture in Azure

- Users login via Azure Entra tenant
- Single page application provides the UI features
- *NGinx Ingress controller with Azure Load balancer* acts as reverse proxy and load balancer.
- *Backend for frontend* handles the application logic
- *Microservices* handle the business-logic, such as ordering or product returns
- All running in Kubernetes pods
- Microservices communicate directly or by events via *EventHubs over Kafka protocol*
- 1 CosmosDb account, multiple databases and collections inside
- Api management integration with AKS

Solution architecture for identity management

- Dedicated environment-specific tenant for the solution.

- Users originate from gmail or MS accounts via OAuth provider
-
- Tenant admin/super users as invited guests from customer and Vincit tenants

- Customer organization also uses Entra tenant for user management
-
- Also reseller users have self-service user management UI for managing organization`s users
  - Implementation via Azure Graph API to call the Azure Entra

RESULTS

- Very satisfied customer & reseller users!
- Major of customer revenue (over 100 Meur) comes via the solution
- Supporting thousands of registered users 24/7 globally with good-enough performance, resilience, security and scalability
- In 6 years only one (1) long downtime because of Azure CosmoDb region being down and fail-over was not working.
-

CHALLENGES

- Managing API:s
  - Migrated from 3rd party API tool to Azure API management
  - In this context - what is “API” and how we manage them between API Gateway and AKS & with Terraform
- AKS maintenance & optimization
  - Finding a balance on cost/performance-optimized AKS node machine type.
  - AKS upgrades
- Managing environments in Azure
  - Via resource groups or accounts?
- Azure Application Gateway vs Nginx ingress controller
  - Application Gateway is costly
- Defining proper role-based access levels in Azure

QUESTIONS?

THANK YOU

Timo Holm
Tech lead / system architect

timo.holm@vincit.fi