

Servisno orijentirana arhitektura SOA

// Osnove SOA
// Arhitektura u praksi
// ESB & Camel
// SOA vs Mikroservisi
// Hrvatsko tržište
// AI & Budućnost

Sadržaj predavanja

01

Osnove i povijest

Što je SOA, definicija, kontekst

02

Arhitektura u dubinu

Načela, sprezanje, ugovori, WSLA/WS-Agreement

03

Otkrivanje servisa

UDDI, WS-Discovery, moderne alternative

04

Kompozicija servisa

Orkestracija, koreografija, BPEL, Saga uzorak

05

SOA vs. Mikroservisi

Prednosti, nedostaci, kada koristiti što

06

SOA u praksi

Globalno, EU razina, hrvatsko tržište, primjeri

07

Evolucija arhitekture

Monolit → ESB → Mikroservisi → API ekonomija

08

AI i budućnost

AI-nativni servisi, agentic orkestracija, MCP



Osnove **i povijest**

Što je SOA?

Servisno orijentirana arhitektura — definicija i kontekst

SOA je pristup razvoju softvera u kojemu su servisi primarni način na koji se poslovna logika čini dostupnom kroz mrežu.

Servis

Samostalna jedinica poslovne funkcionalnosti dostupna mrežom putem standardiziranog sučelja

Labava sprega (*loose coupling*)

Servisi su dizajnirani s minimalnim međuovisnostima — promjena jednog ne kviri druge

Interoperabilnost

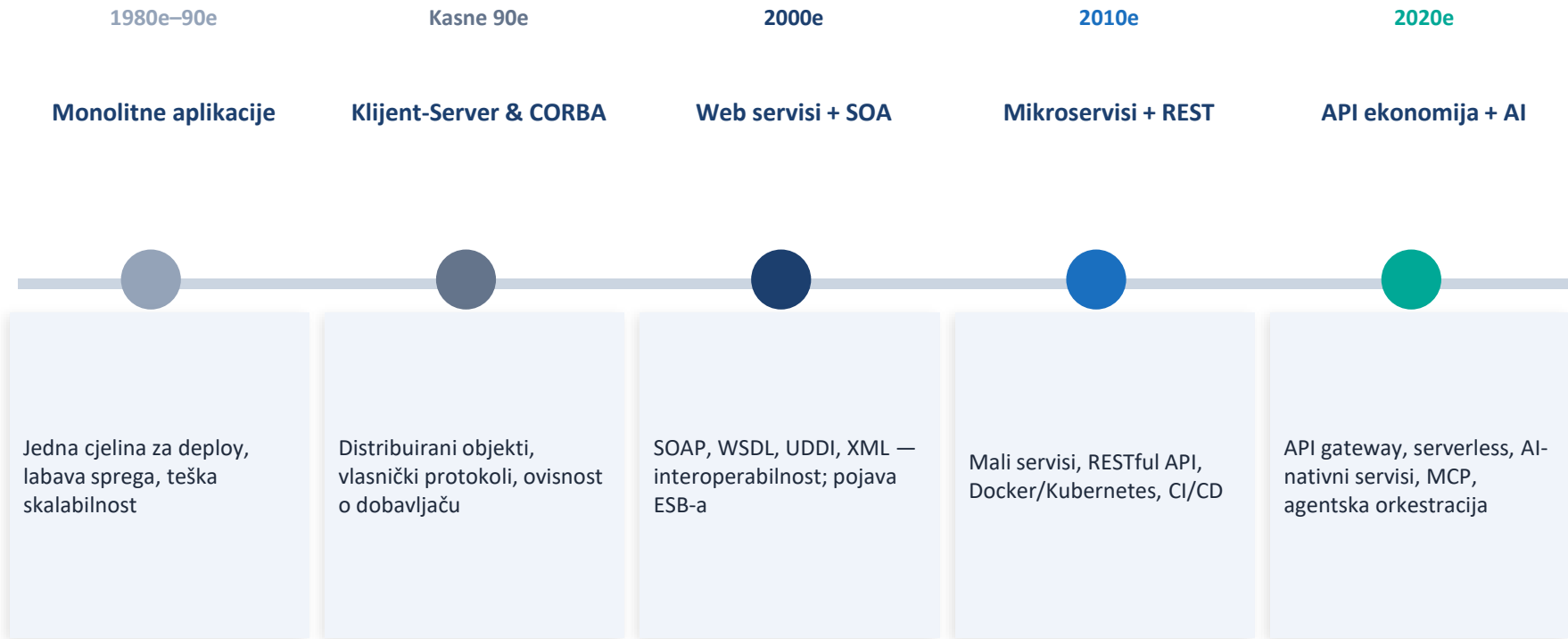
Komunikacija neovisna o platformi putem otvorenih standarda: XML, SOAP, WSDL, HTTP

Mogućnost ponovne uporabe

Jednom izgradi, koristi u više aplikacija i poslovnih procesa

Evolucija arhitekture

Od monolita do inteligentnih servisa



Transformacija: Monolit → Servisi → Mikroservisi

Monolit	SOA + ESB	Mikroservisi
• Jedinstven deployment • Zajednička baza podataka • Tijesno sprezanje modula • Teško skalirati dijelove • Jedan tehnološki stog • Sve ili ništa pri deploymentu	 • Servisi putem mreže • Zajednički ESB sloj • Ugovori (WSDL/SLA) • Ponovna uporaba servisa • Različiti tehnološki stogovi • Orkestrirani procesi (BPEL)	 • Sitnozrnatni servisi • Vlastita baza po servisu • Komunikacija putem API-ja • Neovisan deployment • Polieglotna perzistencija • Nativni CI/CD



Arhitektura

8 načela servisne orijentacije

1 Formalni ugovor

Ugovor definira sučelje, operacije i ulazno/izlazne poruke

2 Labava sprega

Minimizirati direktne ovisnosti između servisa

3 Apstrakcija

Sakriti internu logiku; izložiti samo ugovor (sučelje)

4 Neovisnost

Ograničena logika — svaki servis posjeduje svoju domenu

5 Bez stanja

Servisi ne čuvaju stanje između poziva

6 Mogućnost ponovne uporabe

Dizajnirati za ponovnu uporabu u više scenarija

7 Mogućnost otkrivanja

Jasni, pretraživi opisi u registru

8 Složivost

Servisi mogu biti složeni od manjih servisa

Sprezanje i ugovori servisa

Vrste sprega

Čvrsta sprega

Da bi servisi surađivali, potrebno je poznavati internu strukturu druge strane. Promjena jednog kvari drugi.

Labava sprega

Komunikacija samo putem ugovora. Unutarnje promjene nevidljive su klijentima.

Sadržaj ugovora (SLA/WSDL)

- Formalna definicija sučelja servisa
- Popis operacija koje servis nudi
- Sheme ulaznih / izlaznih poruka (XSD)
- QoS atributi: vrijeme odgovora, dostupnost
- Sigurnosni i autentikacijski zahtjevi
- Pravila verzioniranja i kompatibilnosti
- WSLA (IBM, 2001.) ili WS-Agreement (GGF, 2005.)

Sloj sučelja servisa

SOA uvodi 'treći sloj' između poslovne i aplikacijske logike

Aplikacije se ne spajaju direktno — komuniciraju kroz standardizirani sloj sučelja

Sloj poslovnih procesa

Sloj sučelja servisa

Aplikacija 1
(.NET)

Aplikacija 2
(J2EE)

Aplikacija 3
(Python)

Aplikacija 4
(Legacy)

- Enkapsulira aplikacijsku i poslovnu logiku
- Aplikacije razdvojene prema tehnološkim platformama
- Komponente izložene putem standardnih sučelja
- Omogućuje interoperabilnost heterogenih sustava



Otkrivanje servisa

Otkrivanje servisa: UDDI vs WS-Discovery vs Moderne alternative

UDDI (2000.)

- Univerzalni registar web servisa
- Distribuirani XML registar
- Model Bijelih/Žutih/Zelenih stranica
- API metode: find_business, find_service
- IBM, Microsoft, SAP čvorovi
- Javni registri zatvoreni 2006.
- I dalje prisutan interno u poduzećima

WS-Discovery (2005.)

- OASIS standard, multicast protokol
- Ad-hoc način: probe/resolve poruke
- Upravljeni način: proxy za otkrivanje
- Koristi se u Windowsima (pisači, uređaji)
- SOAP-temeljene poruke
- Nema centralnog registra
- WS-Security za potpisivanje poruka

Moderne alternative

- Consul / etcd (service mesh)
- AWS Service Discovery
- Kubernetes Service DNS
- API gatewayevi (Kong, Apigee)
- OpenAPI / Swagger Hub
- Developer portali (Backstage)
- AI-potpomognuto otkrivanje (2024+)



Kompozicija **servisa**

Orkestracija vs. Koreografija



Orkestracija

- Centralni dirigent (orchestrator) kontrolira tok
- Orchestrator poziva servise redom
- Poslovna logika smještena u orchestratoru
- Jednostavno za centralno praćenje
- Rizik: jedinstvena točka kvara
- Primjer: BPEL, Apache Camel, AWS Step Functions
- WS-BPEL je dominantni XML jezik



Koreografija

- Nema centralnog kontrolera — ravnopravna suradnja
- Svaki servis zna kada treba djelovati (događaji)
- Logika distribuirana po sudionicima
- Otpornija, nema jedinstvene točke kvara
- Teže pratiti cjelokupni tok
- Primjer: Kafka događaji, WS-CDL, ebMS
- Uobičajena u event-driven mikroservisima

Saga *pattern* i distribuirane transakcije

Kako osigurati konzistentnost podataka kroz više servisa bez distribuiranog zaključavanja?



Ako bilo koji korak ne uspije → Kompenzacijske transakcije izvode se obrnutim redoslijedom

Koreografijska Saga

Svaki servis sluša i emitira događaje. Nema centralnog kontrolera. Koristi message brokere (Kafka, RabbitMQ). Dobro se skalira, teže pratiti.

Orkestrirana Saga

Saga orchestrator (npr. AWS Step Functions) eksplicitno govori svakom servisu što napraviti. Lakše vizualizirati i pratiti.

Ostali uzorci

Dvofazni Commit (2PC) — jaka konzistentnost ali loša dostupnost.
Outbox uzorak — jamčena isporuka.
Event Sourcing — revizijski trag putem loga događaja.

Jezici za opis poslovnih procesa

WS-BPEL 2.0

Aktivan (enterprise)

2007.–danas · OASIS standard

XML gramatika za definiranje toka kontrole web servisne orkestracije. Pokreće BPEL engine (Oracle BPEL, Apache ODE). Definira sekvence, paralele, handlere grešaka i kompenzacijsku logiku.

BPML

Zamijenjen

2002. · BPMI.org

Business Process Management Language — utemeljio koncepte koji su utjecali na BPMN. Integriran s WSCI za koreografiju.

BPMN 2.0

Moderni standard

2011.–danas · OMG standard

Business Process Model and Notation — grafički standard za modeliranje procesa. Izvršni BPMN premošćuje jaz između poslovnih analitičara i programera. Koristi se u Camundi, Activitiju, Flowableu.

Apache Camel

Široko korišten

2007.–danas · Apache

Open-source integracijski framework koji implementira Enterprise Integration Pattern (EIP). Usmjerava, transformira i posreduje poruke između sustava putem 300+ konektora.

SOA vs.

Mikroservisi

SOA vs. Mikroservisi: detaljna usporedba

Aspekt	SOA	Mikroservisi
Komunikacija	SOAP/WSDL/ESB	REST/gRPC/Događaji
Veličina servisa	Krupnozrnatna (po domeni)	Sitnozrnatna (po sposobnosti)
Upravljanje podacima	Zajednička baza (često)	Vlastita baza po servisu
Sprezanje	Slabo sprezanje putem ESB-a	Decentralizirano, ravnopravno
Upravljanje	Centralizirano (ESB)	Decentralizirano (API gateway)
Deployment	App server po servisu	Kontejner/serverless
Model tima	Zajednički timovi po sloju	Cross-funkcionalni po servisu
Skaliranje	Skalira cijeli servis	Skalira pojedine funkcije
Najprikladnije za	Integracija enterprise sustava	Cloud-native, brza isporuka

Kada koristiti SOA, a kada Mikroservise?

Odaberite SOA kada...

- Integracija heterogenih enterprise sustava (ERP, CRM, legacy)
- Okruženje s jakim regulatornim zahtjevima koji zahtijevaju formalne ugovore
- Velika organizacija s više poslovnih jedinica koje dijele servise
- Dugoročni, stabilni servisi s mnogo konzumenata
- Tim nema znanja o DevOps-u i kontejnerima
- Potrebno centralizirano praćenje, upravljanje i provedba politika

Odaberite Mikroservise kada...

- Izgradnja cloud-native aplikacija od nule
- Česti ciklusi isporuke (CI/CD) su ključni
- Različiti servisi imaju jako različite zahtjeve za skaliranjem
- Mali, autonomni i cross-funkcionalni timovi
- Polieglotni razvoj — različiti jezici po servisu
- Investicija u infrastrukturu (Kubernetes, observability, API gateway) je moguća

SOA u praksi:

Globalno i lokalno

Tržište SOA — ključni pokazatelji 2024.

**16,1
mlrd\$**

Globalna vrijednost SOA tržišta (2023.)

~8,5%

CAGR prognoza 2024.–2030.

73%

Enterprise tvrtki još uvijek koristi
SOA/ESB

48%

Fortune 500 koristi SOA za integraciju
legacyja

~60%

SOA korisnici koji su prihvatili i
mikroservise

#1

Financijski sektor — najveći usvajatelj
SOA-e

**IBM
Oracle
SAP**

Vodeći SOA platformni dobavljači

ESB

Apache Camel, MuleSoft, WSO2 aktivni
u 2024.

SOA u Europskoj Uniji

Javni sektor, e-Uprava i enterprise integracija

EU e-Uprava i eIDAS

EU uredba eIDAS nalaže interoperabilne digitalne usluge identiteta. Države članice koriste SOA/web servisne arhitekture za prekogranične usluge putem eIDAS mreže čvorova.

EU Direktiva INSPIRE

EU infrastruktura prostornih podataka zahtijeva od vlada izlaganje geospacijalnih skupova podataka kao standardizirani OGC web servisi (WFS, WMS, WCS). Pravi SOA mandat.

EU CEF blokovi

Connecting Europe Facility nudi dijeljene digitalne usluge: eDelivery (AS4 poruke), eID, e-faktura, eSignature — sve temeljeno na SOA načelima sa standardiziranim SOAP/REST sučeljima.

EU Zdravstvo (HL7/IHE)

Prekogranična razmjena zdravstvenih podataka pacijenata putem IHE profila temeljenih na web servisima. Hrvatska sudjeluje u EU eHealth infrastrukturi (eHDSI).

EU Financije (PSD2/GDPR)

PSD2 obvezuje banke na izlaganje Open Banking API-ja — modernu evoluciju SOA-e. GDPR nalaže jasne servisne ugovore s obrađivačima podataka.

EU EIF v2

Europski okvir interoperabilnosti (EIF) v2 eksplicitno promiče servisnu orijentaciju kao temeljnu paradigmu IT sustava javne uprave.

SOA u Hrvatskoj

Javni sektor, bankarstvo i enterprise okruženje

Ključni sektori koji koriste SOA/Web servise

Vlada / e-Građani

Portal e-Građani integrira 40+ vladinih agencija putem web servisa. MUP, FINA, HZMO, HZZO izlažu usluge podataka za građane.

FINA

Hrvatska financijska agencija nudi opsežne B2B/G2B web servisne API-je za e-Račun (e-faktura, obvezatna od 2019.), financijsko izvještavanje i registar poslovanja.

Bankarstvo (HNB/poslovne banke)

HNB i poslovne banke (Erste, Zaba, PBZ, HPBC) koriste SOA za integracije s core bankingom, SEPA plaćanjima i SWIFT konekcijama.

Telekomunikacije (HT, A1, Telemach)

Telco davatelji koriste ESB/SOA za BSS/OSS integraciju — naplatni sustavi, CRM, upravljanje mrežom.

Zdravstvo (CEZIH)

Hrvatski zdravstveni informacijski sustav koristi HL7-temeljene web servise za e-recept, e-uput i razmjenu podataka pacijenata između institucija.

Ključna hrvatska IT infrastruktura

HR CARNET

Akadska mreža — e-učenje, federacija identiteta (eduGAIN, eduroam) putem SAML/WS-Federation

HR OIB sustav

Jedinstveni identifikator građana (OIB) izložen kao nacionalni web servis za sve vladine integracije

HR e-Račun (FINA)

FINA-in portal za e-fakturiranje — obvezatan B2G od 2019., temeljen na XML/web servisima

HR NIAS

Nacionalni sustav autentikacije i autorizacije — federirani identitet za sve e-Uprave servise

Primjeri stvarnih SOA servisa

Servisi koji se i dalje aktivno koriste

Bankarstvo

SWIFT gpi

Global Payments Innovation — SOAP/XML-temeljeno praćenje plaćanja koje koristi 4.000+ banaka u 200 zemalja. Banke izlažu SWIFT web servise za status prekograničnih plaćanja.

e-Trgovina

PayPal SOAP API

PayPal legacy SOAP API i dalje obrađuje milijune transakcija. Merchantice se integriraju putem WSDL ugovora — klasična SOA u produkciji globalnih razmjera.

Logistika

FedEx Web Services

FedEx nudi i SOAP i REST API-je za praćenje. Enterprise klijenti koriste SOAP zbog garantirane stabilnosti ugovora i revizijskih tragova.

Zdravstvo

HL7 FHIR / v2

Health Level 7 — HL7 v2 (SOAP/XML) prisutan je u >80% bolnica. FHIR R4 donosi REST ali koegzistira s legacy SOA sučeljima.

e-Uprava HR

e-Građani

Portal e-Građani agregira 40+ vladinih web servisa. Svaka agencija održava vlastiti SOA endpoint koji se poziva u ime građana.

SAP

SAP ERP Web Services

SAP izlaže tisuće BAPI-ja i IDoca kao SOAP web servise. Svaka veća hrvatska manufacturing ili retail tvrtka koja koristi SAP — koristi SOA.



ESB i evolucija arhitekture

Enterprise Service Bus (ESB)

Integracijska okosnica SOA arhitekture

Aplikacije se spajaju na ESB — ne jedna na drugu. ESB postaje centralni 'mozak' integracije.

SAP ERP

e-Commerce

CRM
(Salesforce)

BI /
Izveštavanje

Legacy
Mainframe

Mobilna
aplikacija

Enterprise Service Bus

Sposobnosti ESB-a:

- Usmjeravanje i transformacija poruka
- Posredovanje protokola (SOAP↔REST↔JMS)
- Orkestracija i izvođenje BPES procesa
- Provedba sigurnosnih politika
- Centralizirano praćenje i logiranje
- Integracija s registrom servisa

ESB platforme: komercijalne i open source

MuleSoft Anypoint

Komercijalna

Najraširenija integracijska platforma. Akvizicija od Salesforcea (6,5 mlrd\$, 2018.). MuleSoft pokreće tisuće enterprise API i SOA integracija.

IBM App Connect

Komercijalna

IBM-ova enterprise ESB/integracijska platforma. Temelj SOA implementacija u Fortune 500 tvrtkama, posebice bankarstvu i osiguranju.

WSO2 Enterprise Integrator

Open Source

Potpuni open-source ESB stog koji podržava SOAP, REST, JMS, datoteke, email, bazu podataka. Popularan u telekomunikacijama, bankarstvu i vladi.

Apache Camel

Open Source

Implementira 300+ Enterprise Integration Patterna. Ugrađen u mnoga prilagođena rješenja. Temelj Red Hat Fuse/Camel-K projekta.

Oracle SOA Suite

Komercijalna

Tijesno integriran s Oracle DB-om, ERP-om i cloud uslugama. Uobičajen u velikim Oracle-stack tvrtkama.

Azure Service Bus + Logic Apps

Cloud

Microsoftov cloud-native ESB ekvivalent. Logic Apps omogućuju vizualnu orkestraciju tijekom rada; Service Bus obrađuje poruke.

AI i

budućnost servisa

AI mijenja što 'servis' uopće znači



LLM kao servis

OpenAI, Anthropic, Google Gemini izlažu AI sposobnosti kao REST/API servise. SOA razmišljanje i dalje vrijedi — ugovori, verzioniranje, SLA.



AI-potpomognuto otkrivanje

Upiti na prirodnom jeziku za pronalazak i kompoziciju servisa. Embedding-temeljena pretraga kataloga API-ja. Alati poput GitHub Copilota sugeriraju pozive API-ja iz konteksta.



Agentska orkestracija

AI agenti (LangChain, AutoGen, CrewAI) dinamički orkestriraju servise u runtime-u na temelju ciljeva — najfleksibilniji oblik koreografije.



Model Context Protocol

MCP (Anthropic, 2024.) standardizira kako AI modeli komuniciraju s alatima i servisima — novi SOA sloj ugovora za AI agente.



Semantički API-ji

API-ji opisani semantičkim metapodacima (ontologije, embeddinzii) omogućuju AI-ju da razmišlja o kompatibilnosti servisa i automatski sastavlja tijekomove rada.



AI-potpomognuti ESB

Platforme sljedeće generacije (IBM watsonx, MuleSoft AI) koriste ML za automapiranje transformacija podataka, predviđanje usmjeravanja i otkrivanje anomalija.

Model Context Protocol (MCP)

Novi SOA sloj ugovora za AI agente

MCP je za AI agente ono što je WSDL bio za SOAP web servise — standardni ugovor za interakciju s uslugom.

Što je MCP?

Otvoreni standard tvrtke Anthropic (2024.). Definira kako AI modeli (Claude, GPT, Gemini) komuniciraju s vanjskim alatima, bazama podataka i servisima. Zamislite: USB-C za AI integracije.

SOA paralela

MCP Server = SOA pružatelj usluge
MCP Klijent = SOA konzument
MCP Alat = SOA operacija
MCP Prompt = SOA ugovor poruke

Ključne sposobnosti

Standardizirani pozivi alata
Pristup resursima (datoteke, baze, API-ji)
Ubacivanje konteksta iz vanjskih izvora
Streaming i async odgovori
Sigurnosni model i dozvole

Stvarni primjeri

MCP serveri za: GitHub, Slack, Google Drive, PostgreSQL, Jira, Salesforce. AI agenti ih otkrivaju i pozivaju kao SOA registre — ali na prirodnom jeziku.

Budući utjecaj

Tvrtke koje grade MCP-kompatibilne SOA servise otključavaju pristup AI agentima. Svaki SOA endpoint s MCP omotačem postaje AI-pozivljivi alat. Koegzistencija legacyja i AI-ja.

Hrvatski kontekst

FINA, e-Građani, CEZIH mogu izložiti MCP omotače nad postojećim web servisima — AI asistenti bi mogli ispunjavati državne obrasce, provjeravati porezni status ili upitati HZZO o osiguranju.

Arhitekturni trendovi 2025.–2030.

Dominacija API ekonomije

API-ji kao proizvodi. Svaki servis je API. OpenAPI, AsyncAPI i GraphQL postaju novi WSDL.

Service Mesh

Istio, Linkerd, Consul Connect — SOA infrastrukturne brige (sigurnost, observability, retry) premještene su u mesh sloj.

Serverless / FaaS

AWS Lambda, Azure Functions — granularnost na razini funkcije. Servisi se komponiraju putem događaja. Nema servera za upravljanje.

AI-nativna integracija

AI agenti koji dinamički otkrivaju, pregovaraju i komponiraju servise. MCP, OpenAPI + LLM-ovi = inteligentni middleware.

Edge Computing

Servisi se izvode na rubu mreže (CDN, IoT gateway). Niskokašnjujuća SOA za scenarije u stvarnom vremenu.

Kvantno sigurni servisi

Post-kvantna kriptografija zahtijeva ponovnu enkripciju svih servisnih ugovora i mehanizama autentikacije.

Ključni zaključci

- 1 SOA ostaje okosnica enterprise integracije — 73% velikih tvrtki i dalje koristi SOA/ESB rješenja uz mikroservise
- 2 Slabo sprezanje, formalni ugovori i otkrivanje servisa su načela koja vrijede bez obzira na arhitekturni stil — SOA, mikroservisi ili AI-nativni pristup
- 3 Hrvatska opsežno koristi SOA: e-Građani, FINA, CEZIH, OIB, NIAS — sve izgrađeno na standardima web servisa; EU usklađenost (INSPIRE, eIDAS, PSD2) nameće SOA načela
- 4 Saga uzorak rješava konzistentnost distribuiranih transakcija bez tijesnog sprezanja — ključno znanje za svakog arhitekta distribuiranih sustava
- 5 Evolucija: Monolit → Web servisi → ESB → Mikroservisi → API ekonomija → AI-nativni servisi — svaka era gradila na lekcijama prethodne
- 6 AI i MCP stvaraju novu SOA renesansu — AI agenti trebaju standardizirane servisne ugovore, mehanizme otkrivanja i uzorke kompozicije, baš kao što je SOA zamišljala 1996.

Pitanja?

**ALGEBRA
BERNAYS
SVEUČILIŠTE**

Servisno orijentirana arhitektura

Interoperabilnost informacijskih sustava — Algebra Bernays Sveučilište 2025/26

Preporučena literatura:

- Thomas Erl — SOA: Principles of Service Design (2007.)
- Sam Newman — Building Microservices (O'Reilly, 2022.)
- Enterprise Integration Patterns — Hohpe i Woolf
- MCP specifikacija — modelcontextprotocol.io
- EU EIF v2 — ec.europa.eu/isa2