

Logic App & Blockchain

Conf. dr. Cristian Kevorchian

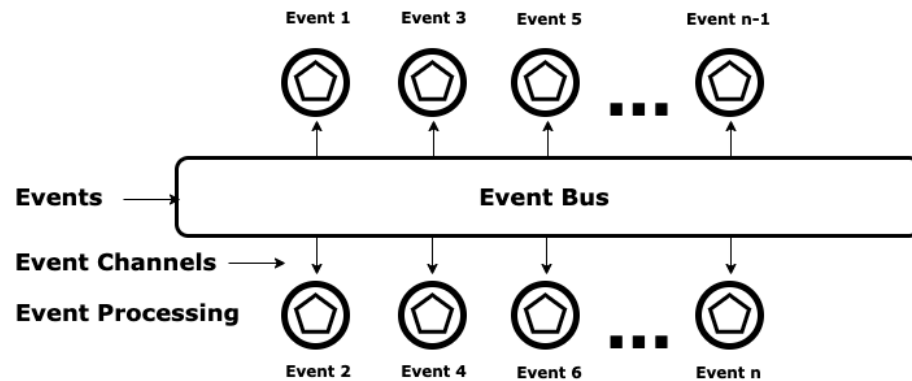
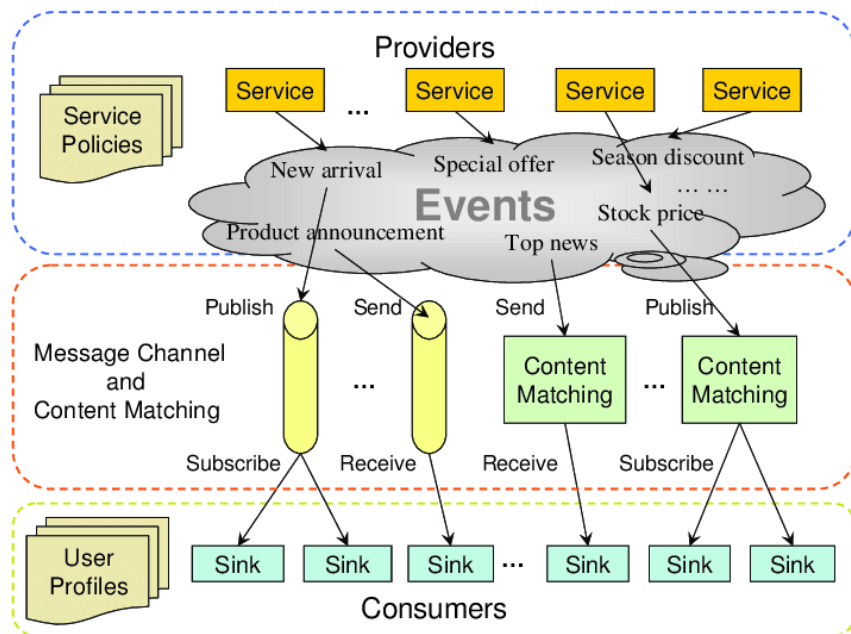
Facultatea de Matematică și Informatică

Evenimente

- ▶ În programare și proiectare software, un **eveniment** este o acțiune recunoscută de un sistem software provenind, de multe ori, din mediul extern și care poate fi gestionată de sisteme software.
- ▶ Evenimentele de natură computațională pot fi generate sau declanșate de sistem, de utilizator sau în diverse alte moduri.
- ▶ De obicei, evenimentele sunt tratate sincron cu fluxul de programe; adică software-ul poate avea unul sau mai multe noduri de procesare a evenimentelor.
- ▶ O sursă de evenimente include:
 - ▶ utilizatorul, care poate interacționa cu software-ul prin perifericele computerului - de exemplu, tastând pe tastatură.
 - ▶ un dispozitiv hardware,
 - ▶ software-ul poate declanșa propriul set de evenimente în ciclul de viață al evenimentului, de ex. pentru a comunica finalizarea unei sarcini. Software-ul care își schimbă comportamentul ca răspuns la evenimente devine interactiv.

Arhitecturi Bazate pe Evenimente

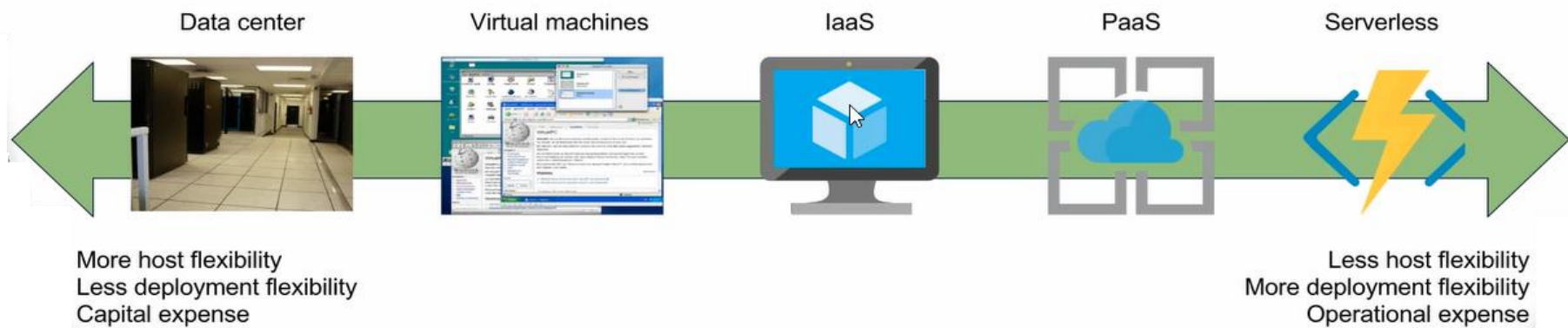
- ▶ Arhitectura bazată pe evenimente (ABE) este o paradigmă arhitecturală în dezvoltarea de software care promovează producția, detectarea, consumul și reacția la evenimente.
- ▶ Acest model arhitectural poate fi implementat prin proiectarea și implementarea de aplicații și sisteme care transmit evenimente între componentele și serviciile software slab cuplate.
- ▶ Un sistem bazat pe evenimente se compune din: emițători de evenimente, consumatori de evenimente și canale de evenimente.
 - ▶ Emitenții au sarcina de a detecta, colecta și transfera evenimente. Emitenții nu cunosc consumatorii evenimentului, neștiind dacă există un consumator și, în cazul în care există, nu știu cum va fi utilizat ulterior evenimentul.
 - ▶ Consumatorii trebuie să expună o reacție de îndată ce este prezentat un eveniment. Reacția ar putea sau nu să fie complet asigurată de consumator. De exemplu, consumatorul ar putea avea doar responsabilitatea de a filtra, transforma și transmite evenimentul către o altă componentă sau ar putea oferi o reacție autonomă la un astfel de eveniment.
 - ▶ Canalele de evenimente sunt canale prin care acestea sunt transmise de la emitenți la consumatorii de evenimente. Distribuția corectă a evenimentelor este atributul exclusiv al canalului de evenimente. Implementarea fizică a canalelor de evenimente se poate baza pe componente tradiționale, cum ar fi un middleware orientat pe mesaje sau o comunicare punct-la-punct, care ar putea necesita un cadru tranzacțional mai elaborat.



Arhitecturi bazate pe evenimente

Arhitecturi pentru sisteme distribuite

- ▶ Dezvoltarea de aplicații distribuite având drept fundament arhitecturi bazate pe evenimente simplifică scalabilitatea orizontală în modelele de calcul distribuit și în plus înregistrează și o reziliență superioară.
- ▶ Acest lucru se datorează faptului că starea aplicației poate fi replicată pe mai multe instantante paralele pentru a realiza o disponibilitate ridicată.
- ▶ Evenimentele noi pot fi inițiate oriunde, dar, mai important, acestea se propagă prin rețeaua de “data store” actualizându-le pe măsură ce sosesc.
- ▶ Adăugarea de noduri suplimentare devine banală. Un snapshot al aplicației, preia un flux de evenimente și și rulează cu acesta.
- ▶ Arhitectura bazată pe evenimente poate completa arhitectura orientată pe servicii (SOA), deoarece serviciile pot fi activate prin declanșatoare care se execută la evenimentele de intrare. Această paradigmă este deosebit de utilă ori de câte ori consumatorul nu oferă niciun executabil autonom.



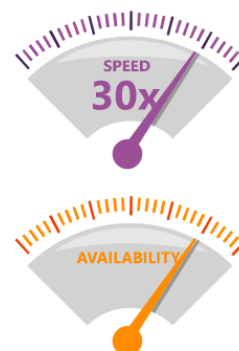
Evoluția platformelor pentru aplicații

Serverless manifesto

- ▶ **Function** sunt unități de deployment și scalare.
- ▶ **Scalare per request** Utilizatorii nu pot provisiona capacitatea nici în sus și nici în jos.
- ▶ **Nu se plătește niciodată pentru inactivitate** (fără servere / containere reci sau costurile acestora)
- ▶ **Codul de invocare bazat pe trigger** rulat fiind din cauza apariției unui eveniment care trebuie tratat



Abstractizarea
serverelor



Bazat-pe-Evenimente/
Scalare în Timp-Real



Granularitatea
Taxării

Implementări: Azure Serverless

Există trei elemente pe care se fundamentează Azure Serverless



Functions

Procesarea evenimentelor cu ajutorul codului serverless



Logic Apps

Automatizarea accesului și utilizarea datelor multi tenant cu un volum mic de cod scris



Event Grid

Prelucrarea unor volume mari de evenimente

Serverless

Development

 IDE support

 Integrated DevOps

 Local development

 Monitoring

 Visual Debug History

Platform

 Functions

- ✓ Developer productivity
- ✓ Triggers and Bindings
- ✓ Flexible deployment options

 Logic apps

- ✓ Visual designer
- ✓ 100+ connectors
- ✓ Functions orchestration

 Event Grid

- ✓ Manage all events in one place
- ✓ Near real-time delivery
- ✓ Broad coverage

Database



Storage



Security & Access Control



IoT



Analytics



Intelligence



Azure Logic Apps

Unice în Logic Apps

- ▶ Implementare și orchestrare de fluxuri de lucru destinate integrării concepuale
- ▶ Organizarea microserviciilor distribuite
 - ▶ Sau funcțiilor
 - ▶ Fiecare acțiune este un [http endpoint](#)
- ▶ Conectori - Conectorii oferă acces rapid de la Azure Logic Apps la evenimente, date și acțiuni în alte aplicații, servicii, sisteme, protocoale și platforme. Prin utilizarea conectorilor în aplicațiile logice se extind capacitățile pentru aplicațiile cloud și locale pentru a efectua task-uri cu datele pe care le creați sau aveți deja stocate.
- ▶ Declarative JSON în WDL(Workflow Definition Language)

Logic Apps Connectors (120+)

Cloud APIs and platform functionality

- Over 120 built-in connectors
- Hosted and managed within the platform
- Scales to meet your needs
- First class designer experience
- Rapid development

API connections

- Authenticate once and reuse
- Differentiate connection configuration
- Simple to deploy
- Portal experience for managing API Connections

SaaS

- appFigures
- Asana

- Azure API Management
- Azure App Services
- Azure Automation
- Azure Cognitive Face API
- Azure Cognitive LUIS
- Azure Cognitive Text Analytics
- Azure Cognitive Vision
- Azure Data Lake Store
- Azure Document DB
- Azure Event Hub
- Azure Functions
- Azure Machine Learning
- Azure Resource Manager
- Azure Service Bus
- Azure SQL
- Azure Storage Blob
- Azure Storage Queues

- Basecamp
- Bing Search

- BitBucket
- Bitly
- Blogger
- Box
- Buffer
- Campfire
- Chatter

- Common Data Service
- Disqus
- DocuSign
- Dropbox

- Dynamics AX Online
- Dynamics CRM Online
- Dynamics CRM Service Bus

- Dynamics Financials
- Dynamics Operations

- Easy Redmine
- Eventbrite
- Facebook
- FreshBooks
- Freshdesk
- GitHub
- Gmail
- Google Calendar
- Google Contacts
- Google Drive
- Google Sheets
- Google Tasks
- GoTo Meeting
- GoTo Training
- GoTo Webinar
- Harvest
- HelloSign
- Infusionsoft
- JIRA
- Insightly
- Instagram
- Instapaper
- MailChimp
- Mandrill
- Medium

- Microsoft Project Online
- Microsoft Translator
- MSN Weather
- Muhimbi PDF

- Office 365
- Office 365 Users
- Office 365 Video
- OneDrive

- OneDrive for Business
- OneNote
- Outlook.com
- Outlook Tasks

- PagerDuty
- Pinterest
- Pipedrive
- Pivotal Tracker

- Power BI
- Project Online

- Redmine
- Salesforce
- Salesforce Chatter
- SendGrid

- SharePoint Online
- Slack
- SmartSheet
- SparkPost
- Stripe
- Survey Monkey
- Todoist
- Toodledo
- Trello
- Twilio
- Twitter
- Typeform
- UserVoice

- VS Team Services
- Webmerge
- Wordpress

- Wunderlist
- Yammer
- YouTube
- Zendesk

Protocols/native

- HTTP, HTTPS
- HTTP Webhook
- FTP, SFTP
- SMTP
- RSS
- Compose, Query, Parse JSON
- Wait
- Terminate
- Workflow

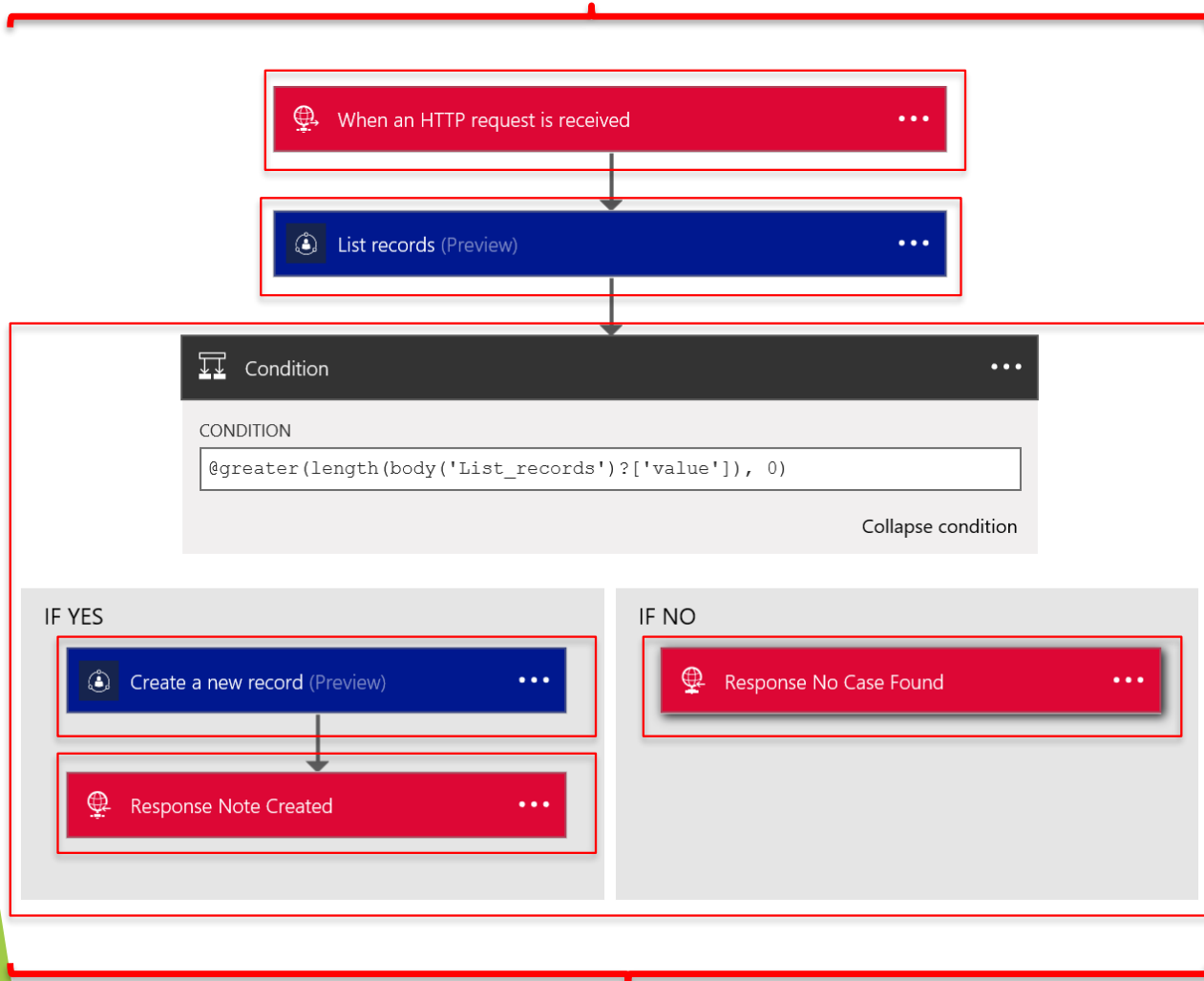
XML and EDI

- XML Validation
- Transform XML (+Mapper)
- Flat File Encode
- Flat File Decode
- X12
- EDIFACT
- AS2
- Integration Account Artifact Lookup

Hybrid

- BizTalk Server
- File System
- IBM DB2
- Informix
- Oracle DB
- SharePoint Server
- SQL Server
- SAP
- Websphere MQ

Concepte



- ▶ Controlul flow-ului prin intermediul condițiilor, buclelor, tratarea excepțiilor etc.
- ▶ Pattern-uri multiple
 - ▶ Invocare manuală
 - ▶ Bazată pe evenimente
 - ▶ Operații programate
- ▶ Workflow-uri
- ▶ Trigger-e
- ▶ Acțiuni
- ▶ API-uri gestionate
- ▶ Structuri de flow-uri

WDF(Workflow Definition Language)

- ▶ “Code Behind”
- ▶ Structură de baza

```
{  
  "$schema": "<schema-of the-definition>",  
  "contentVersion": "<version-number-of-definition>",  
  "parameters": { <parameter-definitions-of-definition> },  
  "triggers": [ { <definition-of-flow-triggers> } ],  
  "actions": [ { <definition-of-flow-actions> } ],  
  "outputs": { <output-of-definition> }  
}
```

Azure Event Grid

- Azure Event Grid este un serviciu cloud care furnizează infrastructură pentru EDC(Event-Driven Computing), agnostic la orice limbaj sau platformă.
- Event Grid se concentrează pe evenimente sau mesaje care indică faptul că „s-a întâmplat ceva”.
- Serviciul permite trimiterea de mesaje, direcționarea către endpoint-uri și consum-ul cu piese de cod personalizate.
- Permite livrarea aproape în timp real (de obicei mai puțin de o secundă) volume mari de evenimente.

Anatomia Azure Event Grid

Evenimente – Ceea ce se întâmplă

Evenimente sources/publishers – Unde evenimentul are loc
















Topici – Un canal pentru evenimente conexe

Subscriptions(eventimente) – Un endpoint s-au mecanism built-in de rutare a evenimentelor, uneori catre multiple handler-e.

Event handler-e - App sau serviciu care reactionează la eveniment.

Filtre – Pot fi aplicate unor tipuri de evenimente sau evenimente legate de subiect(upload de fisiere .txt)

Event Sources

	Azure Blob Storage
	Azure resource groups
	Azure subscriptions
	Azure Event Hubs
	Azure Media Services
	Azure IoT Hub
	Azure Service Bus
	Azure Maps
	Azure Container Registry
	Azure SignalR Service
	Azure App Configuration
	Azure Machine Learning
	Azure Communication Services
	CloudEvents Sources
	Custom Events (anything)

Event Grid



Event Handlers

Serverless Code



Functions

Workflow and Integration



Service Bus



Logic Apps

Buffering and Competing Consumers



Event Hubs



Storage Queues

Other Services and Applications



Hybrid Connections
(WebSockets)



WebHooks
(anything)

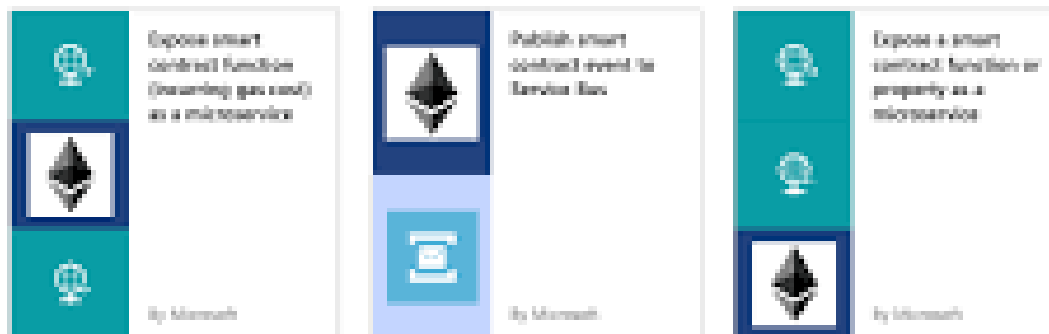


Azure
Automation

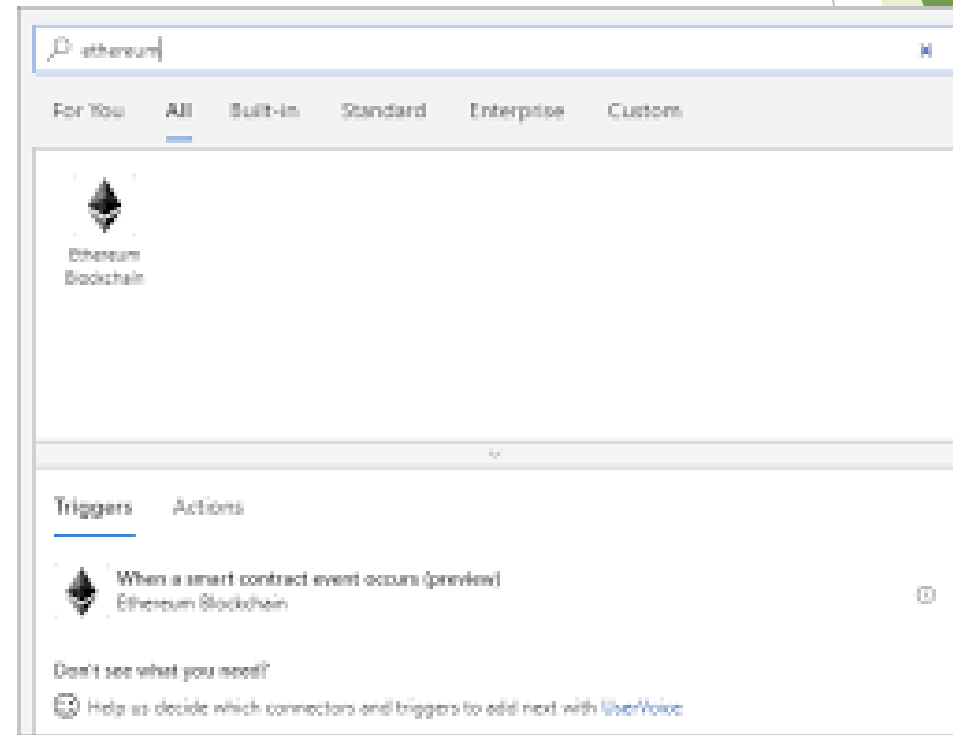
Ethereum Blockchain (Preview)

Ethereum is a smart contract blockchain. With the Ethereum Blockchain connector, you can perform actions such as submitting transactions, reading contract state, and trigger flows on contract updates.

[See the documentation.](#)



Apelarea Contractelor Inteligente în Azure Logic App



DEMO Conectori Ethereum