A Cognitive Serverless Framework for the Cloud-Edge Continuum

D3.6 COGNIT FaaS Model - Software Source - a

Version 1.0

31 October 2023

Abstract

COGNIT is an AI-enabled Adaptive Serverless Framework for the Cognitive Cloud-Edge Continuum that enables the seamless, transparent, and trustworthy integration of data processing resources from providers and on-premises data centers in the cloud-edge continuum, and their automatic and intelligent adaptation to optimise where and how data is processed according to application requirements, changes in application demands and behaviour, and the operation of the infrastructure in terms of the main environmental sustainability metrics. This document offers a catalogue of those open source software resources developed in WP3 “Distributed FaaS Model for Edge Application Development” during the First Research & Innovation Cycle (M4-M9) as part of the implementation of several of the main components of the COGNIT Framework (i.e. Device Client, Serverless Runtime, and Provisioning Engine).

This project is funded by the European Union’s Horizon Europe research and innovation programme under Grant Agreement 101092711 – SovereignEdge.Cognit

Deliverable Metadata

Project Title:	A Cognitive Serverless Framework for the Cloud-Edge Continuum
Project Acronym:	SovereignEdge.Cognit
Call:	HORIZON-CL4-2022-DATA-01-02
Grant Agreement:	101092711
WP number and Title:	WP3. Distributed FaaS Model for Edge Application Development
Nature:	R: Report
Dissemination Level:	PU: Public
Version:	1.0
Contractual Date of Delivery:	30/09/2023
Actual Date of Delivery:	31/10/2023
Lead Author:	Idoia de la Iglesia (Ikerlan)
Authors:	Monowar Bhuyan (UMU), Malik Bouhou (CETIC), Aritz Brosa (Ikerlan), Sébastien Dupont (CETIC), Torsten Hallmann (SUSE), Johan Kristiansson (RISE), Martxel Lasa (Ikerlan), Marco Mancini (OpenNebula), Alberto P. Martí (OpenNebula), Philippe Massonet (CETIC), Nikolaos Matskanis (CETIC), Daniel Olsson (RISE), Michał Opala (OpenNebula), Goiuri Peralta (Ikerlan), Samuel Pérez (Ikerlan), Thomas Ohlson Timoudas (RISE), Paul Townend (UMU), Iván Valdés (Ikerlan), Constantino Vázquez (OpenNebula).
Status:	Submitted

Document History

Version	Issue Date	Status[1]	Content and changes
0.1	20/10/2023	Draft	Initial Draft
0.2	27/10/2023	Peer-Reviewed	Reviewed Draft
1.0	31/10/2023	Submitted	Final Version

Peer Review History

Version	Peer Review Date	Reviewed By
0.1	27/10/2023	Marco Mancini (OpenNebula)
0.1	27/10/2023	Paul Townend (UMU)

Summary of Changes from Previous Versions

First Version of Deliverable D3.6

[1] A deliverable can be in one of these stages: Draft, Peer-Reviewed, Submitted, and Approved.

Executive Summary

This is the first version of Deliverable D3.6, the COGNIT FaaS Model Software Source report, produced in WP3 “Distributed FaaS Model for Edge Application Development”. It provides a short description, licence, version, code repository and user guide, as well as design, testing, and verification reference of each of the software requirements that have had active development tasks during the First Research & Innovation Cycle (M4-M9) in connection with these main components of the COGNIT Framework:

Device Client

- **SR1.1** Interface with Provisioning Engine:

  Implementation of the communication with the Provisioning Engine.

- **SR1.2** Interface with Serverless Runtime:

  Implementation of the communication of with the Serverless Runtime

- **SR1.3** Programming languages:

  Support for different programming languages.

Serverless Runtime

- **SR2.1** Secure and Trusted FaaS Runtimes:

  Automated building of secure and trusted images (vulnerability scans, security assessment) related to different flavours of FaaS Runtimes.

Provisioning Engine

- **SR3.1** Provisioning Interface for the Device to manage Serverless Runtimes:

  Provide an interface to the Device asking for a Serverless Runtime to offload functions and data transfer on any resource of the cloud-edge continuum.

This deliverable has been released at the end of the First Research & Innovation Cycle (M9), and will be updated with incremental releases at the end of each research and innovation cycle (i.e. M15, M21, M27, M33).

Table of Contents

Abbreviations and Acronyms

AI	Artificial Intelligence
API	Application Programming Interface
CC	Confidential Computing
CD	Continuous Delivery
DaaS	Data as a Service
DB	Database
FaaS	Function as a Service
GPU	Graphics Processing Unit
HTTP	Hypertext Transfer Protocol
IAM	Identity and Access Management system
IOPS	I/O Operations Per Second
IP	Internet Protocol
IoT	Internet of Things
JSON	Javascript Object Notation
LDAP	Lightweight Directory Access Protocol
ML	Machine Learning
NIS	Network and Information Security
OIDC	OpenID Connect
OS	Operating System
QoS	Quality of Service
REST	Representational State Transfer
RBAC	Role-Based Access Control
S3	Simple Storage Service
SDK	Software Development Kit
SEV	Secure Encrypted Virtualization
SGX	Software Guard eXtension
SLA	Service Level Agreement
SQL	Structured Query Language
TEE	Trusted Execution Environments
TLS	Transport Layer Security
VM	Virtual Machine
YAML	Yaml Ain’t a markup language

1. Device Client

SR1.1 Interface with Provisioning Engine	
Description	The Device Client is the component that enables the devices to communicate with the COGNIT Framework in order to perform the offloading of tasks. This component communicates with the Provisioning Engine to request/delete/update a Serverless Runtime. It communicates with the provided Serverless Runtime to perform the offloading of functions and the uploading of content to the Data Service, if configured. The device runtime is delivered as a library with implementations in Python and C which abstracts the user from the internal application protocol by offering a user-friendly API. The interface with the Provisioning Engine establishes communication with the COGNIT Framework, allowing the device to access its permitted resources.
Licence	Apache 2.0
Version	e8e4336
Design	D3.1 → [SR1.1] Interface with Provisioning Engine
Code	Public Repository
User Guide	Repository README
Testing	D5.2 → 10.1 Device Client
Verification	D5.2 → 10.1 Device Client

SR1.2 Interface with Serverless Runtime	
Description	The interface with the Serverless Engine allows the user interacting with the Serverless Runtime to which it has been assigned. Through the defined API, the Device Client is able to manage offloaded tasks at the convenience of the application it is running.
Licence	Apache 2.0
Version	e8e4336
Design	D3.1 → [SR1.2] Interface with Serverless Runtime

Code	Public Repository
User Guide	Repository README
Testing	D5.2 → 10.1 Device Client
Verification	D5.2 → 10.1 Device Client

SR1.3 Programming languages	
Description	In this version only the Python version of the Device Client has been implemented (representing interpreted languages).
Licence	Apache 2.0
Version	e8e4336
Design	D3.1 → [SR1.3] Programming languages
Code	Public Repository
User Guide	Repository README
Testing	D5.2 → 10.1 Device Client
Verification	D5.2 → 10.1 Device Client

2. Serverless Runtime

SR2.1 Secure and Trusted FaaS Runtimes	
Description	The Serverless Runtime is the service deployed into the scheduled node that will be in charge to execute the offloaded tasks. This service exposes the Serverless Runtime API to allow the devices to upload the functions and the needed data to execute them. There will be several flavours of Serverless Runtime to be deployed, depending on the function requirements. It will communicate through the defined RESTful API with the Device Client that is offloading the concerned function. The Serverless Runtime's image will need to contain all the software requirements for the function to be executed. This requirement focuses on the FaaS (Function as a Service), the actual environment where the offloaded function will be executed.
Licence	Apache 2.0
Version	8b4bea5
Design	D3.1 → [SR2.1] Secure and Trusted FaaS Runtimes
Code	Public repository
User Guide	Repository README
Testing	D5.2 → 10.2 Serverless Runtime
Verification	D5.2 → 10.2 Serverless Runtime

3. Provisioning Engine

SR3.1 Provisioning Interface for the Device to manage Serverless Runtimes	
Description	The Provisioning Engine is a software component that acts as the single point of contact for any device / application that requests access to a Serverless Runtime. It consists of a FaaS Runtime to offload computation through the FaaS paradigm, and/or a DaaS Runtime to offload data into the cloud. Once this component receives a request for a Serverless Runtime, it communicates with the Cloud-Edge Manager, waits for the Serverless Runtime to be available, and returns the endpoints for the Device Runtime to communicate with it.
Licence	Apache 2.0
Version	1bec3ee7
Design	D3.1 → [SR3.1] Provisioning Interface for the Device to manage Serverless Runtimes
Code	Public Repository
User Guide	Repository Wiki
Admin Guide	Repository Wiki
Testing	D5.2 → 10.3 Provisioning Engine
Verification	D5.2 → 10.3 Provisioning Engine