

# OpenFHE

OpenFHE: Open-Source  
Fully Homomorphic Encryption Library  
November 7, 2022

---

Ahmad Al Badawi  
[contact@openfhe.org](mailto:contact@openfhe.org)

# OPENFHE DESIGN PRINCIPLES

- OpenFHE, a new open-source C++17 Fully Homomorphic Encryption (FHE) software library that incorporates selected design ideas from prior FHE projects, including **PALISADE**, **HElib**, **HEAAN**, and **FHEW**, and includes several new design concepts and ideas.
- The main new design features can be summarized as follows:
  - From the **cryptography** perspective, we assume from the very beginning that all implemented FHE schemes will support **bootstrapping** and **scheme switching**
    - Common features are used by the implementation, for example, the key switching implementation is shared by multiple Single Instruction, Multiple Data (SIMD) schemes
  - From the **performance** perspective, OpenFHE supports multiple hardware acceleration backends using a standard **Hardware Abstraction Layer (HAL)**
  - From the **usability** perspective, OpenFHE includes both
    - **user-friendly modes**, where all maintenance operations, such as modulus switching, key switching, and bootstrapping, are automatically invoked by the library, and
    - **compiler-friendly modes**, where an external compiler makes these decisions

# CRYPTOGRAPHIC CAPABILITIES

- OpenFHE includes efficient implementations of all common FHE schemes:
  - Brakerski/Fan-Vercauteren (BFV) scheme for integer arithmetic
  - Brakerski-Gentry-Vaikuntanathan (BGV) scheme for integer arithmetic
  - Cheon-Kim-Kim-Song (CKKS) scheme for real-number arithmetic
    - with approximate bootstrapping
  - Ducas-Micciancio (DM/FHEW) and Chillotti-Gama-Georgieva-Izabachene (CGGI/TFHE) schemes for Boolean circuit evaluation
    - with arbitrary-function evaluation for larger plaintext moduli
- OpenFHE also includes the following multiparty (MP) extensions of FHE:
  - Threshold FHE for BGV, BFV, and CKKS schemes
  - Proxy Re-Encryption (PRE) for BGV, BFV, and CKKS schemes

# KEY FACTS ABOUT OPENFHE

- Preview release (v0.9) launched on July 19, 2022
  - A stable version (v1.0) was released on Nov 3<sup>rd</sup>
- Designed by (some of) authors of PALISADE, HELib, HEAAN, and FHEW libraries
- Official successor of PALISADE
- Complies with the HomomorphicEncryption.org post-quantum security standards for homomorphic encryption
- We offer OpenFHE under the 2-clause Berkeley Software Distribution (BSD) open-source license, making it easier to wrap and redistribute OpenFHE in products
- Generously supported by DARPA
- A community-driven open-source project developed by a diverse group of contributors from both industry and academia, including Duality, Samsung, Intel, MIT, UCSD, and others
- Google Transpiler uses the CGGI (TFHE) implementation as the FHE backend
- OpenFHE is formally affiliated with the NumFocus stable of open-source software projects

# DESIGN PAPER [<https://eprint.iacr.org/2022/915>]

Paper 2022/915

## OpenFHE: Open-Source Fully Homomorphic Encryption Library

*Ahmad Al Badawi*, Duality Technologies

*Jack Bates*, Duality Technologies

*Flavio Bergamaschi*, Intel Corporation

*David Bruce Cousins*, Duality Technologies

*Saroja Erabelli*, Duality Technologies

*Nicholas Genise*, Duality Technologies

*Shai Halevi*, Algorand Foundation

*Hamish Hunt*, Intel Corporation

*Andrey Kim*, Samsung Advanced Institute of Technology

*Yongwoo Lee*, Samsung Advanced Institute of Technology

*Zeyu Liu*, Duality Technologies

*Daniele Micciancio*, University of California, San Diego, Duality Technologies

*Ian Quah*, Duality Technologies

*Yuriy Polyakov*, Duality Technologies

*Saraswathy R.V.*, Duality Technologies

*Kurt Rohloff*, Duality Technologies

*Jonathan Saylor*, Duality Technologies

*Dmitriy Suponitsky*, Duality Technologies

*Matthew Triplett*, Duality Technologies

*Vinod Vaikuntanathan*, Massachusetts Institute of Technology, Duality Technologies

*Vincent Zucca*, DALI, Universite de Perpignan Via Domitia, LIRMM, University of Montpellier

# SCHEME SUPPORT MATRIX

Library/ Scheme or Extension	BGV	BGV Bootstr.	BFV	CKKS	CKKS Bootstr.	DM	CGGI	Threshold FHE (MP)	PRE (MP)
Concrete							✓		
<b>FHEW</b>						✓			
<b>HEAAN</b>				✓	✓				
<b>HELib</b>	✓	✓		✓					
Lattigo			✓	✓	✓			✓	
<b>OpenFHE</b>	✓	*	✓	✓	✓	✓	✓	✓	✓
<b>PALISADE</b>	✓		✓	✓		✓	✓	✓	✓
SEAL	✓		✓	✓					
TFHE							✓		

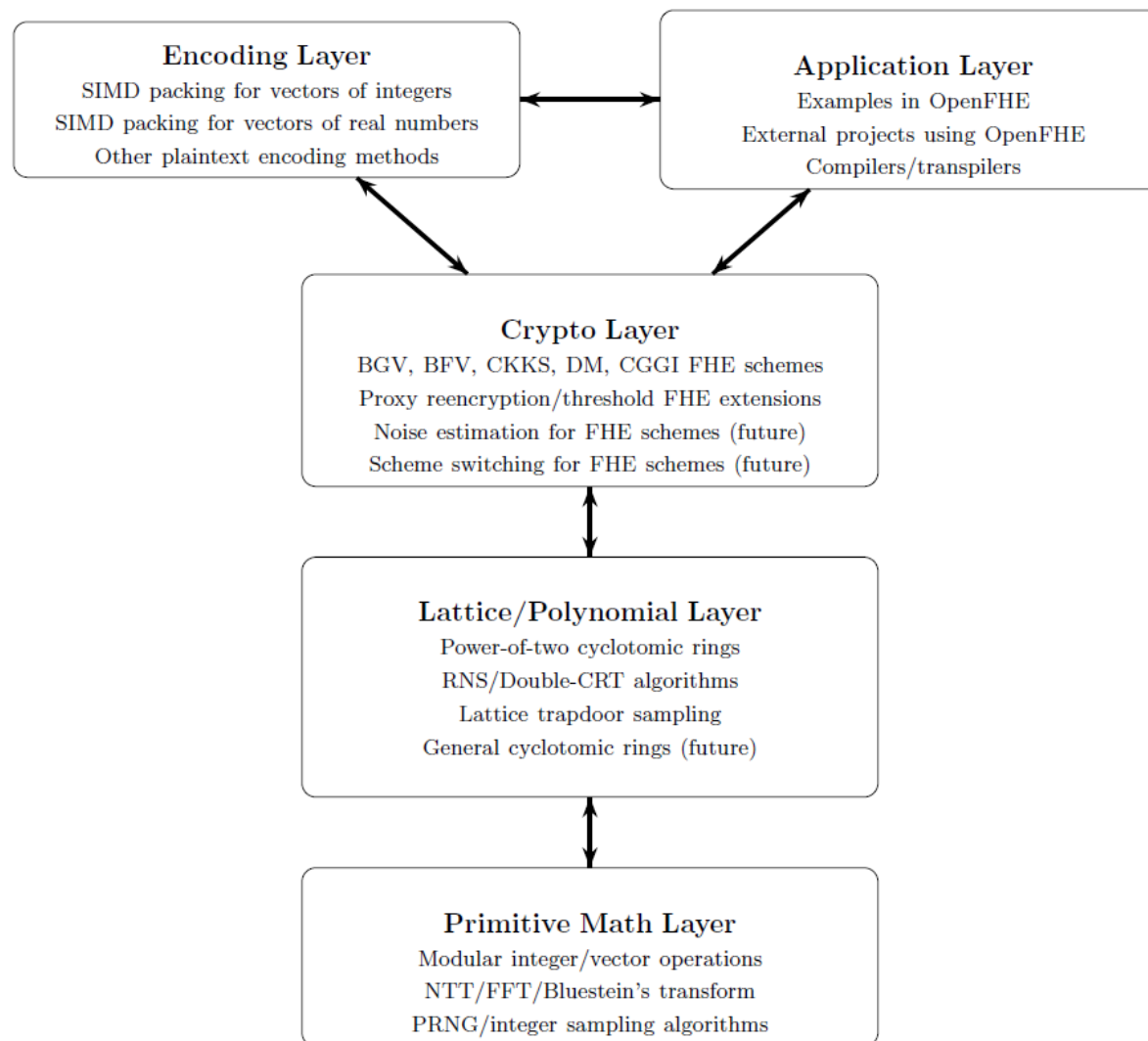
\* - prototype exists, but not part of release

# SUMMARY OF NEW FEATURES IN OPENFHE

Includes all prior FHE functionality of PALISADE. Also adds the following new features:

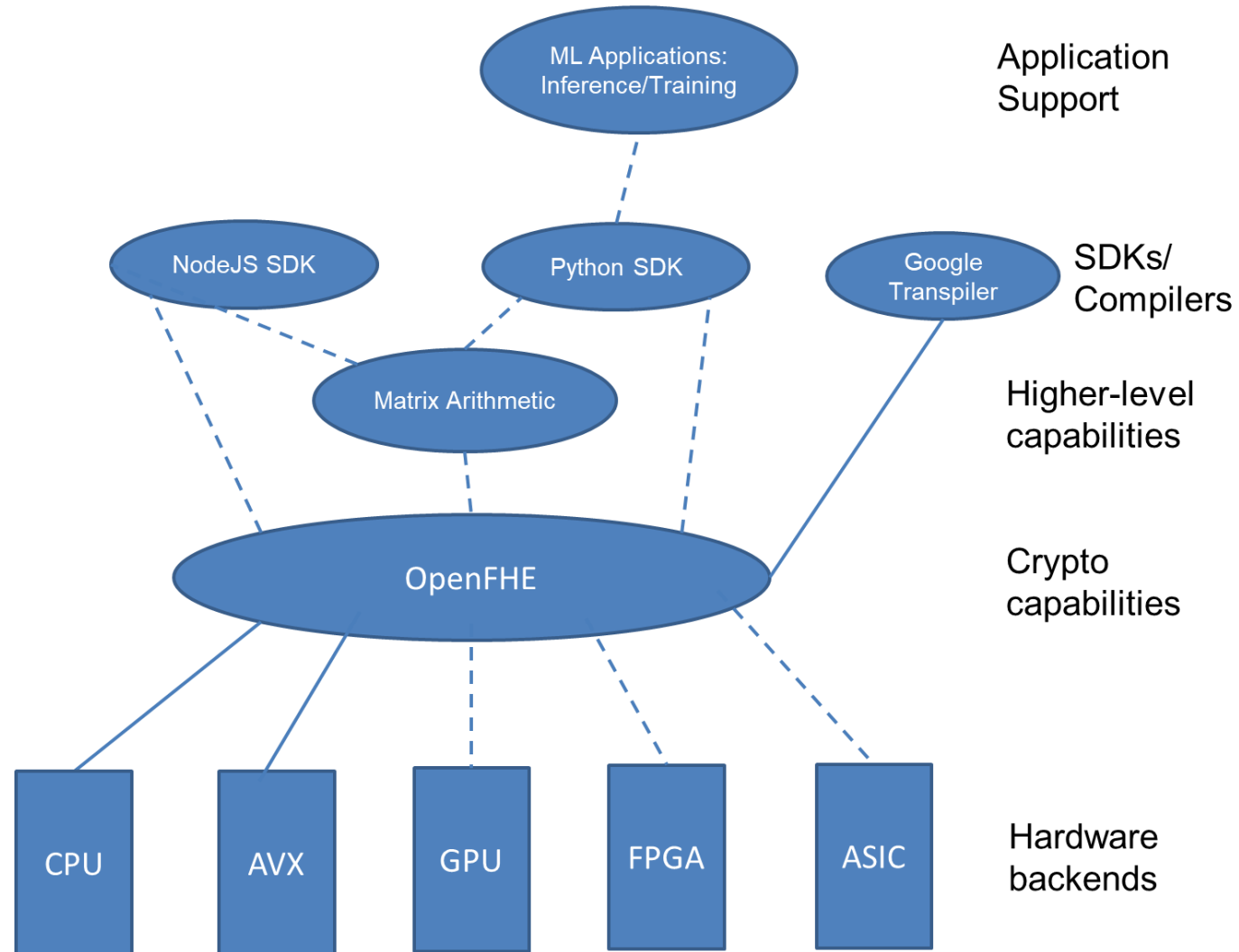
- New BGV and BFV Residue Number System (RNS) variants proposed in <https://eprint.iacr.org/2021/204>
- A new CKKS RNS variant proposed in <https://eprint.iacr.org/2020/1118>
- A full RNS implementation of CKKS bootstrapping (important for deep learning!)
- Large-precision comparison and other algorithms based on functional bootstrapping, which are proposed in <https://eprint.iacr.org/2021/1337>
- Adds support for multiple hardware acceleration backends using a Hardware Abstraction Layer feature
  - Intel HEXL library implemented as a backend for CPUs with AVX-512 extensions

# LAYERS IN OPENFHE (CONTRIBUTOR VIEW)





# BROADER OPENFHE COMMUNITY (USER VIEW)



SDK: Software Development Kit; ML: Machine Learning; CPU: Central Processing Unit; AVX: Advanced Vector eXtensions; GPU: Graphic Processing Unit; FPGA: Field-Programmable Gate Array; ASIC: Application-Specific Integrated Circuit

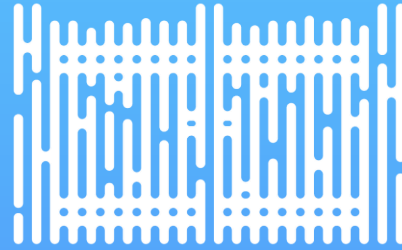
Distribution Statement A – Approved for Public Release, Distribution Unlimited

# OPENFHE VISION FOR MACHINE LEARNING (ML) USING FHE

- The main ML focus is on
  - Approximate method based on CKKS
  - Hybrid approximate/LUT approach based on CKKS and DM (FHEW) /CGGI (TFHE)
- Features that are already available in OpenFHE
  - CKKS bootstrapping to support deep learning
  - Large-precision comparison and small-precision LUT evaluation
- Features under development
  - A prototype of scheme switching from CKKS to/from DM/CGGI already exists, and this feature is expected to be available in OpenFHE shortly after the first stable release
  - Matrix arithmetic library
  - Python SDK
  - NodeJS SDK

# MAIN RESOURCES AND LINKS FOR OPENFHE

- OpenFHE design paper: <https://eprint.iacr.org/2022/915>
- OpenFHE website: <https://openfhe.org>
- ReadTheDocs documentation for OpenFHE: <https://openfhe-development.readthedocs.io/en/latest/>
- OpenFHE development repository: <https://github.com/openfheorg/openfhe-development>
- OpenFHE github organization where various OpenFHE-dependent projects are housed: <https://github.com/openfheorg>
- Community Forum for OpenFHE: <https://openfhe.discourse.group/>
- Acknowledgements: This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.



# THANK YOU

[contact@openfhe.org](mailto:contact@openfhe.org)