HEaaN2
A High-Performance CKKS Library

Presented by Seonghak Kim on the behalf of Crypto Lab. Inc.

Core Dev Team : S. Jeon, M. Kim, S. Kim, J. Lee, J. Lee, T. Noh, Erkhes N.

Former Members : J. Ju, S. Jung, S. Kim, J. Mono, E. Suvanto

HEAAN
CRYPTO LAB

CRYPTO LAB
주식회사 크립토랩

TABLE
OF
CONTENTS

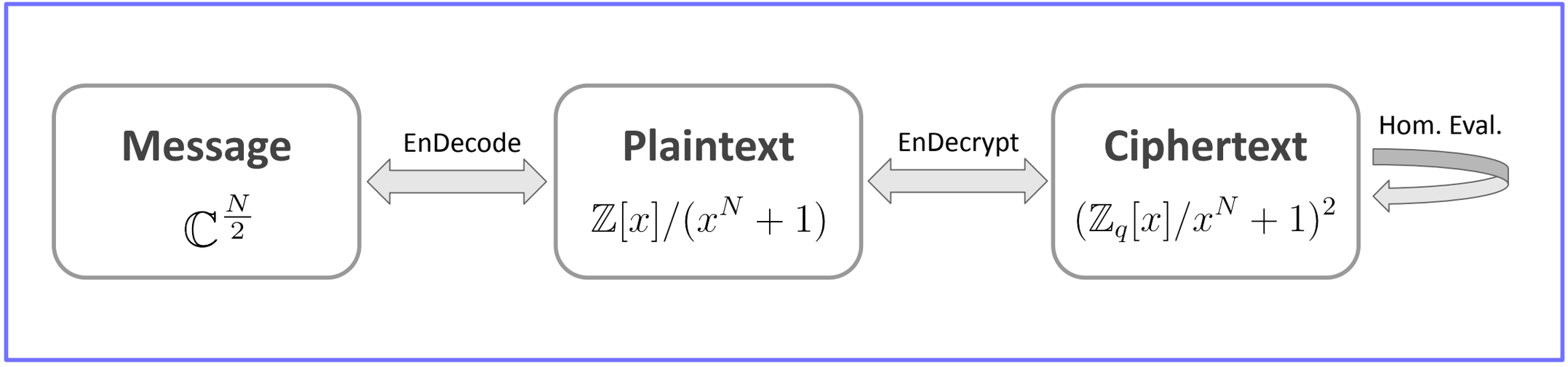
Overview of HEaaN2

Overview of HEaaN2

- ***CKKS only.***
- ***Closed-source.***
- ***SOTA performance.***
- ***Available at*** : heaan.io

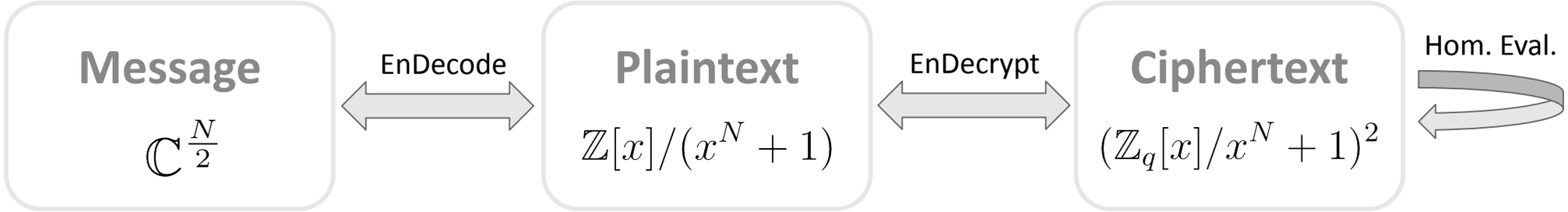
Overview of HEaaN2

Basic CKKS



Overview of HEaaN2

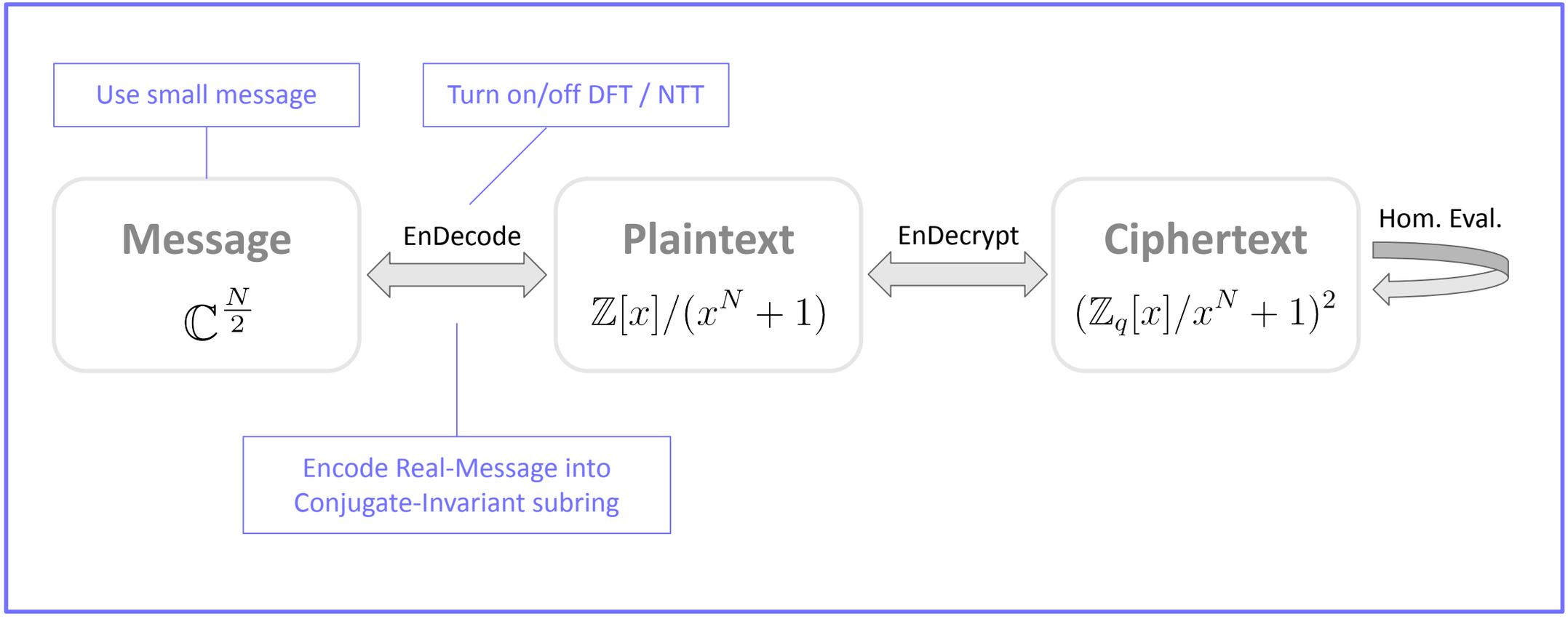
Extended CKKS for High Performance



Goal : Optimize This!

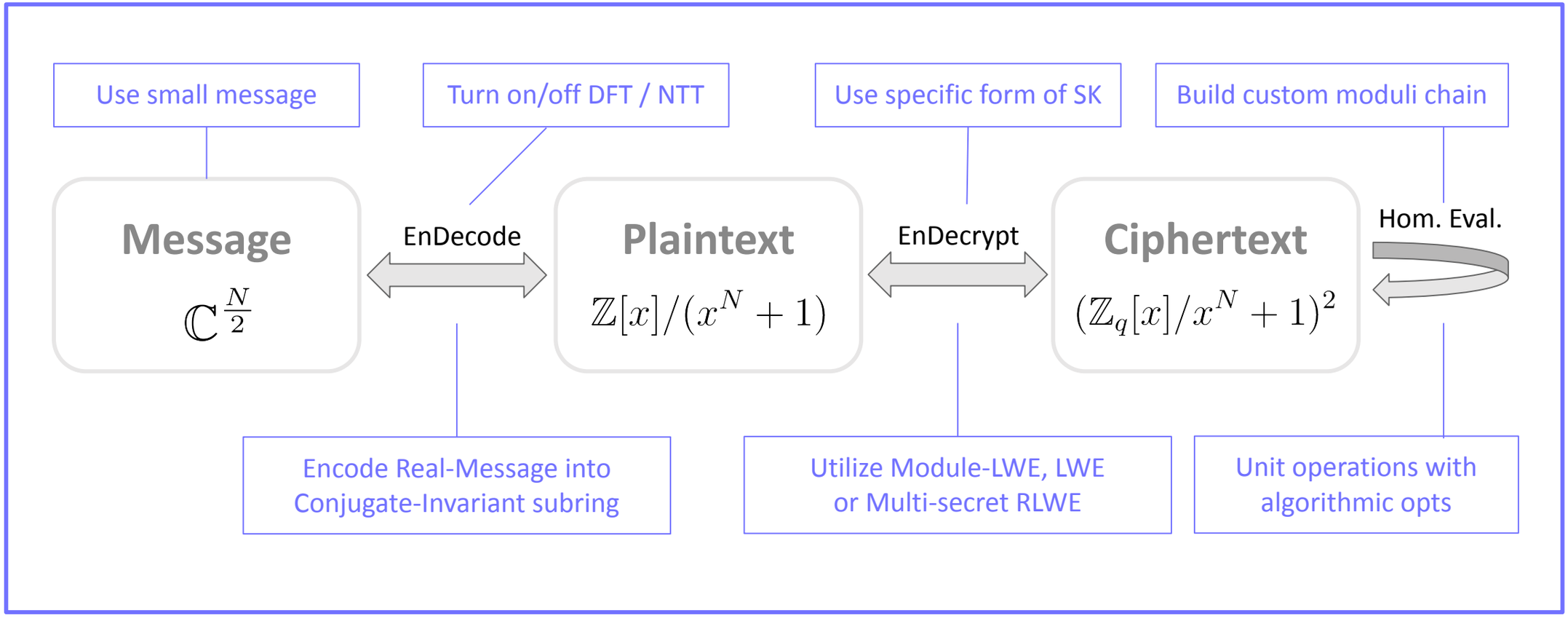
Overview of HEaaN2

Extended CKKS for High Performance

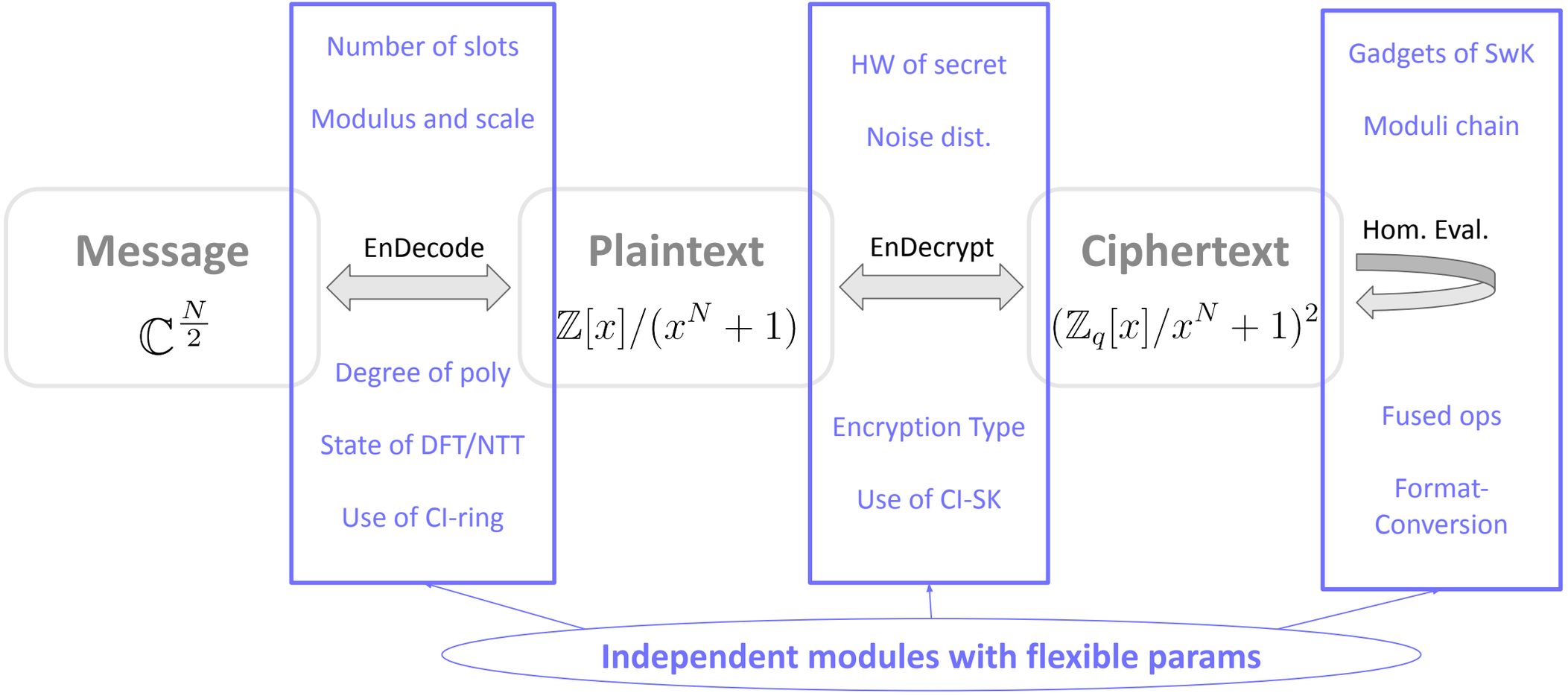


Overview of HEaaN2

Extended CKKS for High Performance



Overview of HEaaN2



Overview of HEaaN2

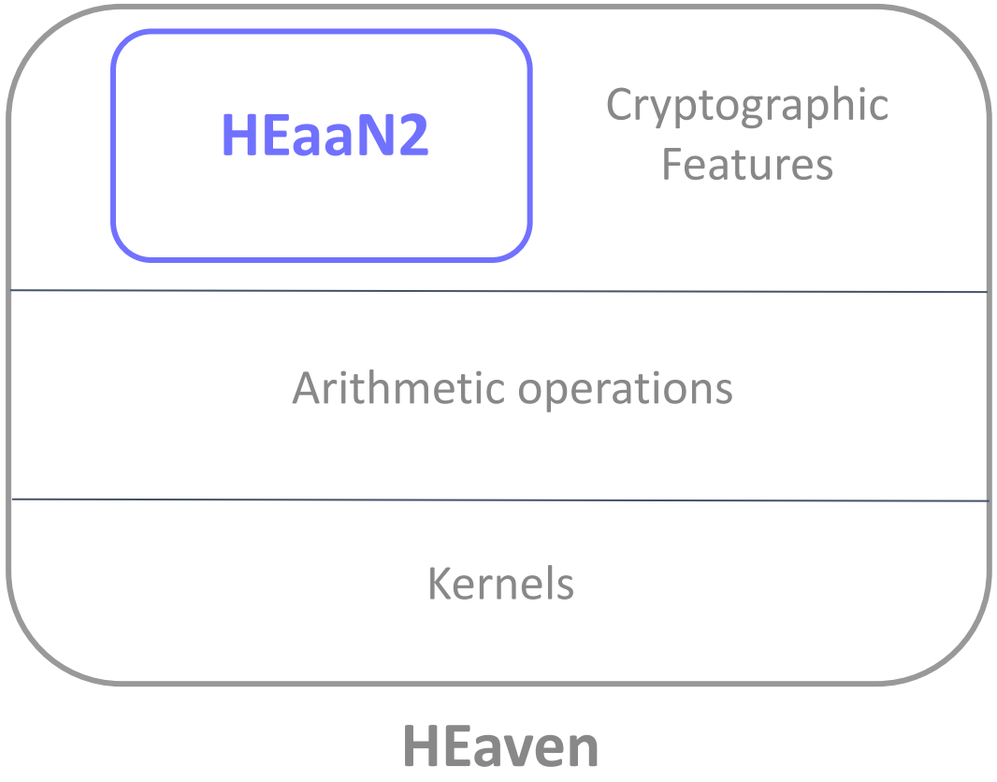
CPU, 16 threads	OpenFHE	HEaaN2	Ratio
Addition	3.85 ms	0.09 ms	**42.8**x
Multiplication	121.4 ms	10.69 ms	**11.4**x
Bootstrap	11,341 ms	665 ms	**17.1**x
Levels gained by BTS	4	15	**3.75**x

GPU(RTX 5090)	Cheddar	HEaaN2 v0.1.1	Theodosian	HEaaN2 v0.2.0
Bootstrap	23.18 ms	14.17 ms	12.75 ms	**9.62** ms

GPU(A100)	NVIDIA Cerium	HEaaN2 v0.1.1	HEaaN2 v0.2.0
Bootstrap	34.2 ms	29.1 ms	**20.9** ms

OpenFHE : v1.4.2
CPU : INTEL XEON GOLD 6544Y @3.6GHz
All parameters are logN=16, logPQ < 1730, prec > 18 bit,
Details adjusted to be disavantageous on HEaaN2.

Overview of HEaaN2



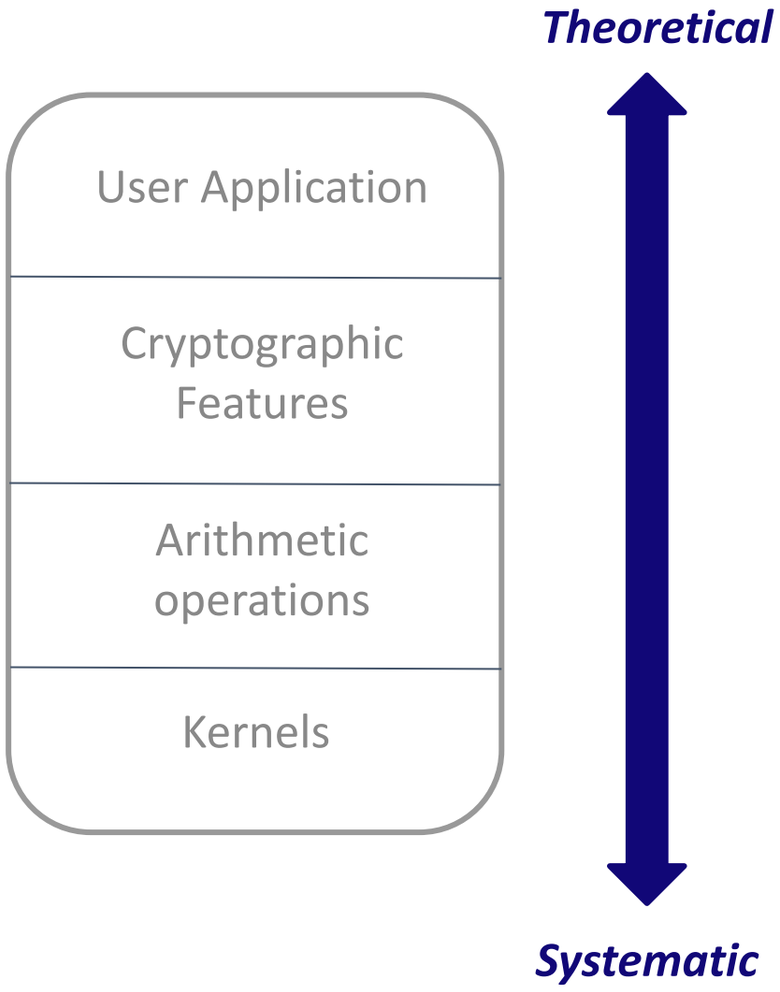
- In this version, we have :
  - Grafting
  - Custom parameters
  - Flexible moduli chain & scale
  - Conjugate-Invariant subring
  - Batching

- In future, we may release :
  - LWE, MLWE, Ring packing
  - Multi-secret RLWE
  - High-precision FHE
  - Bootstrapping on discrete data
  - 32bit-word support
  - ***More to come!***

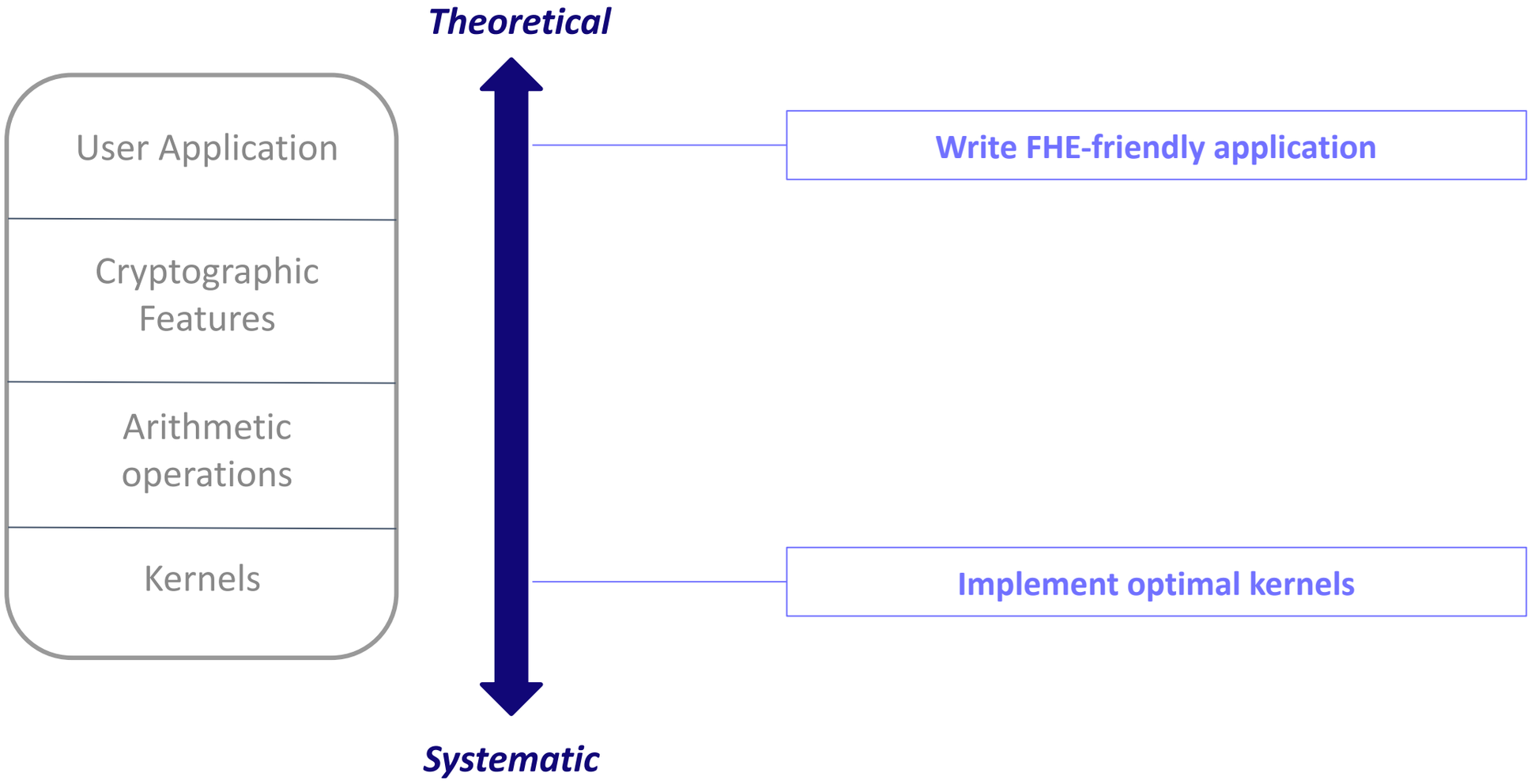
CKKS Engineering :

Performance vs Abstraction

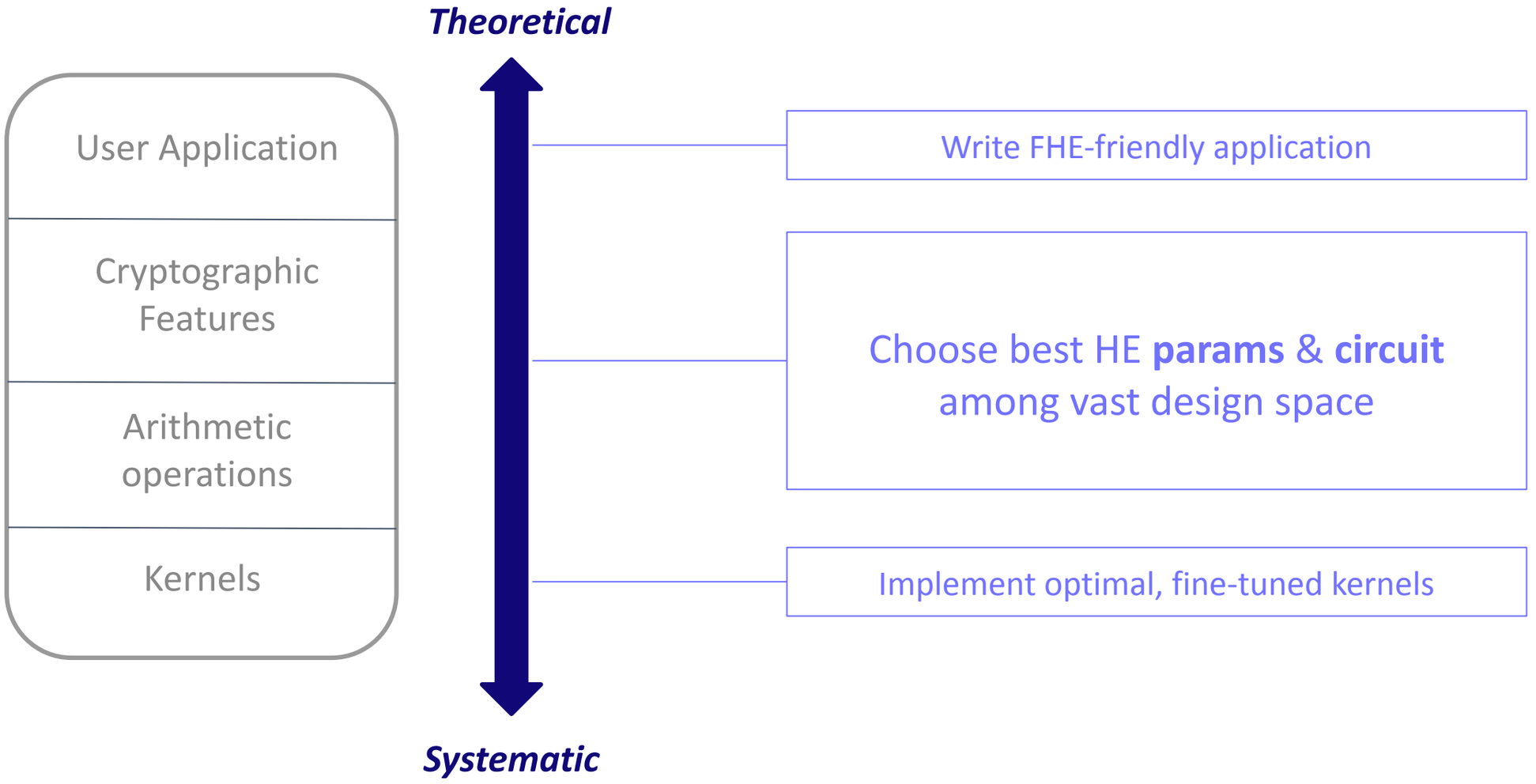
CKKS Engineering : Performance vs Abstraction



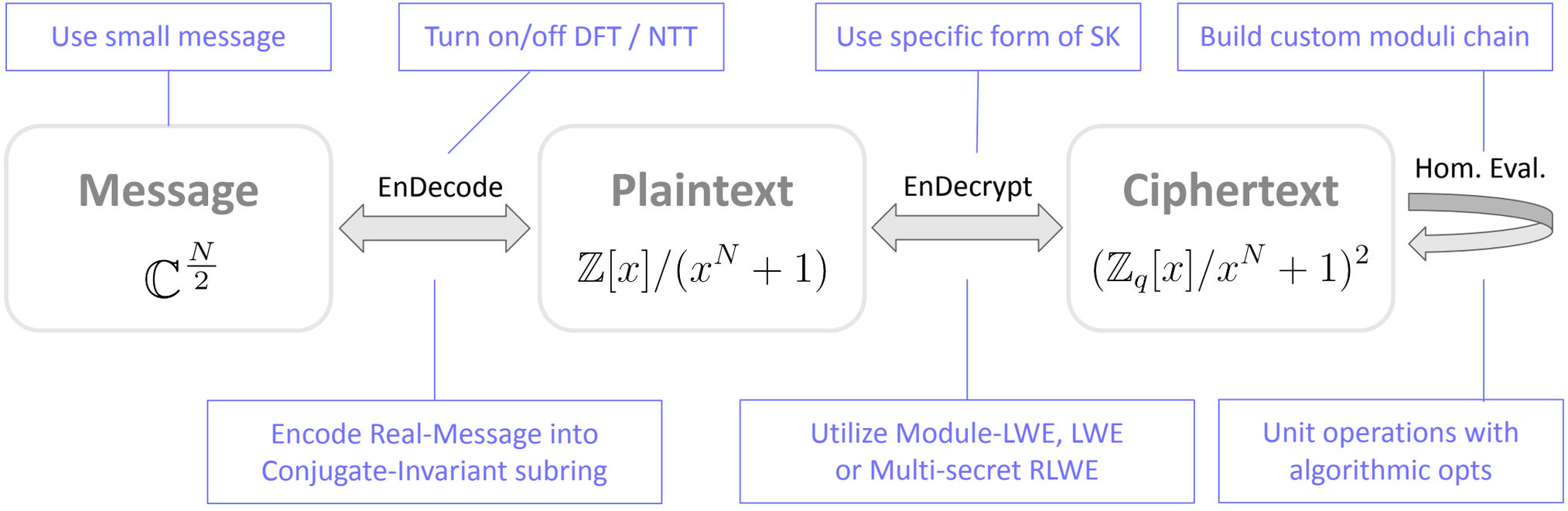
CKKS Engineering : Performance vs Abstraction



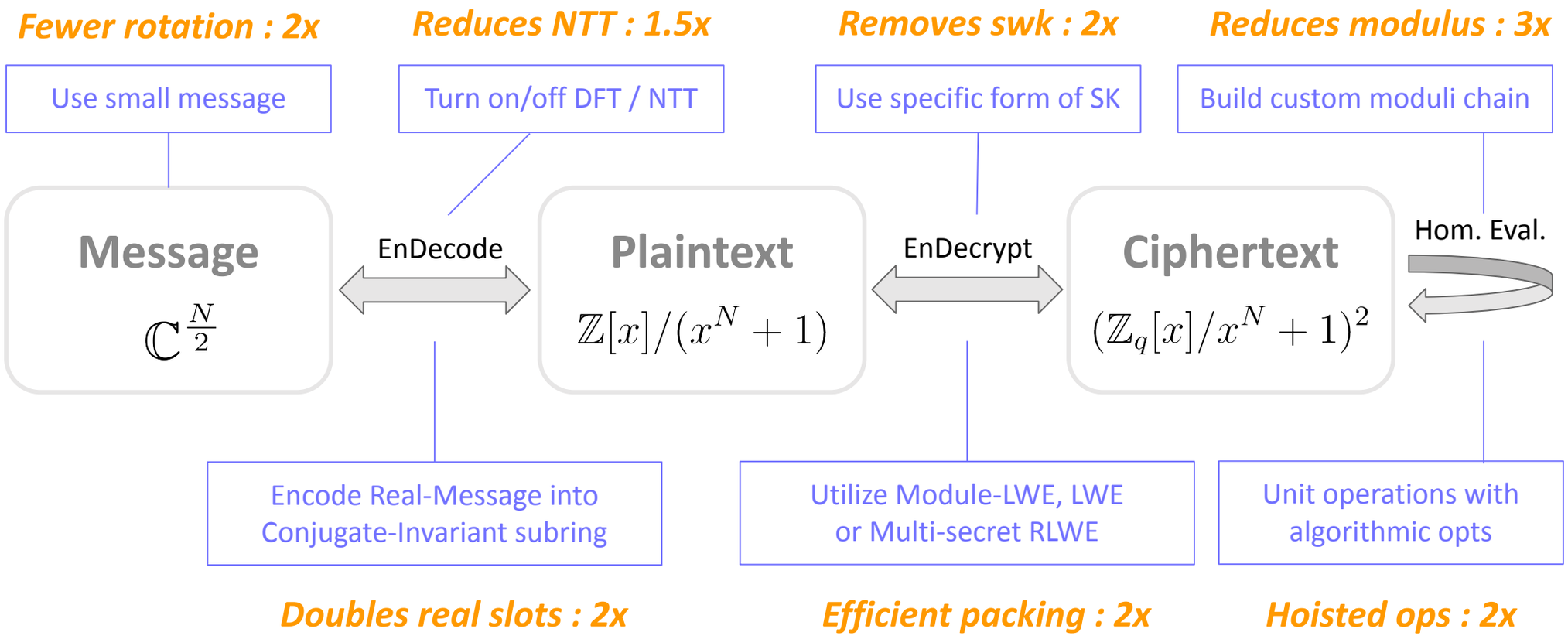
Performance vs Abstraction



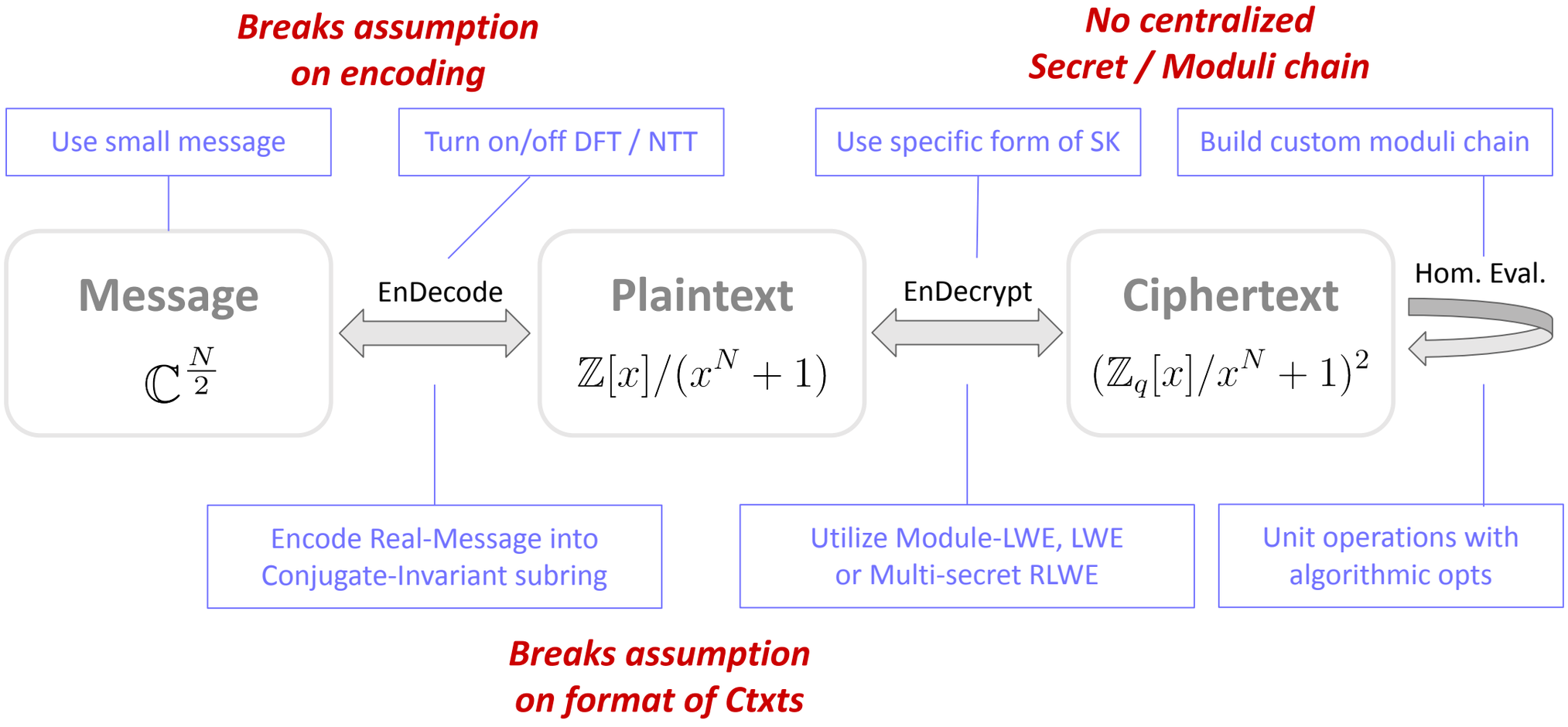
CKKS Engineering : Performance vs Abstraction



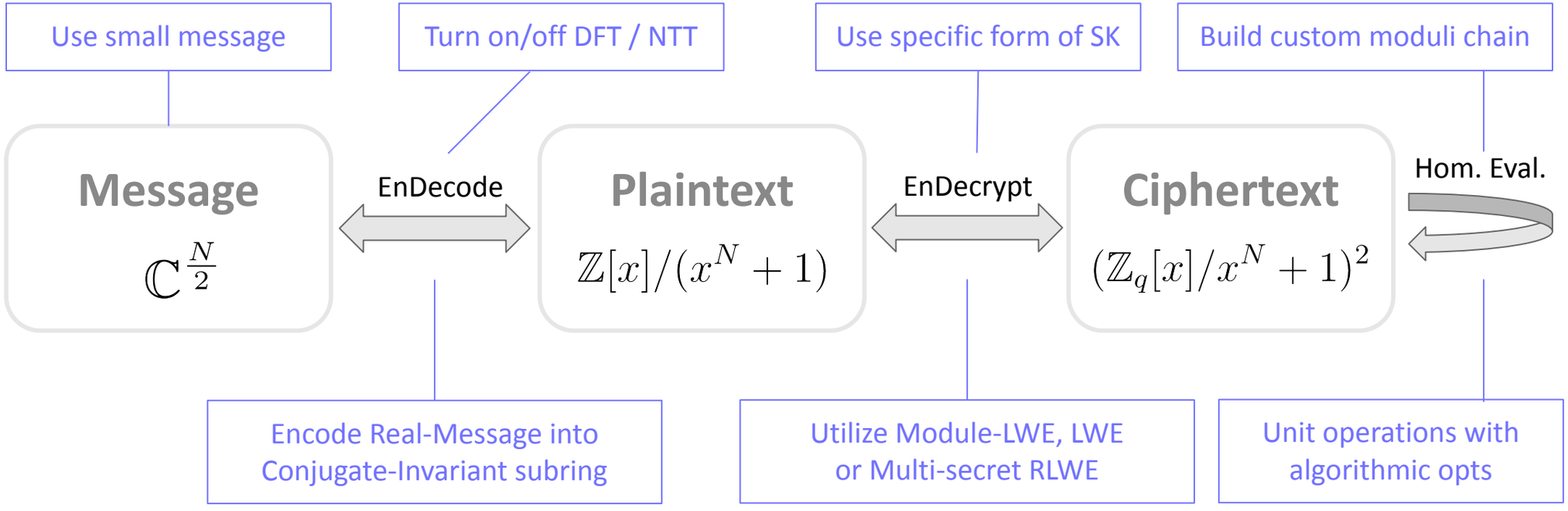
CKKS Engineering : Performance vs Abstraction



CKKS Engineering : Performance vs Abstraction



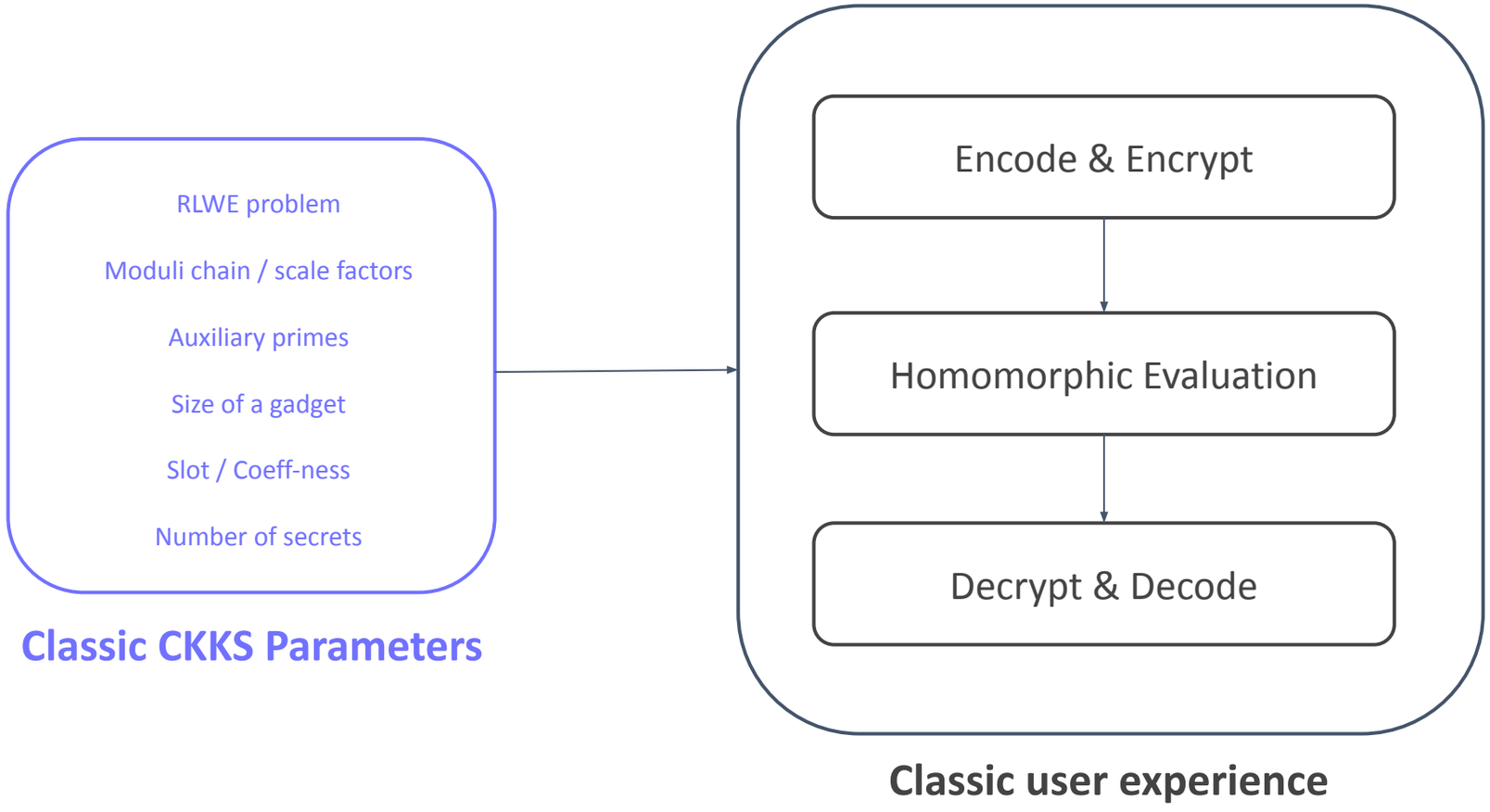
CKKS Engineering : Performance vs Abstraction



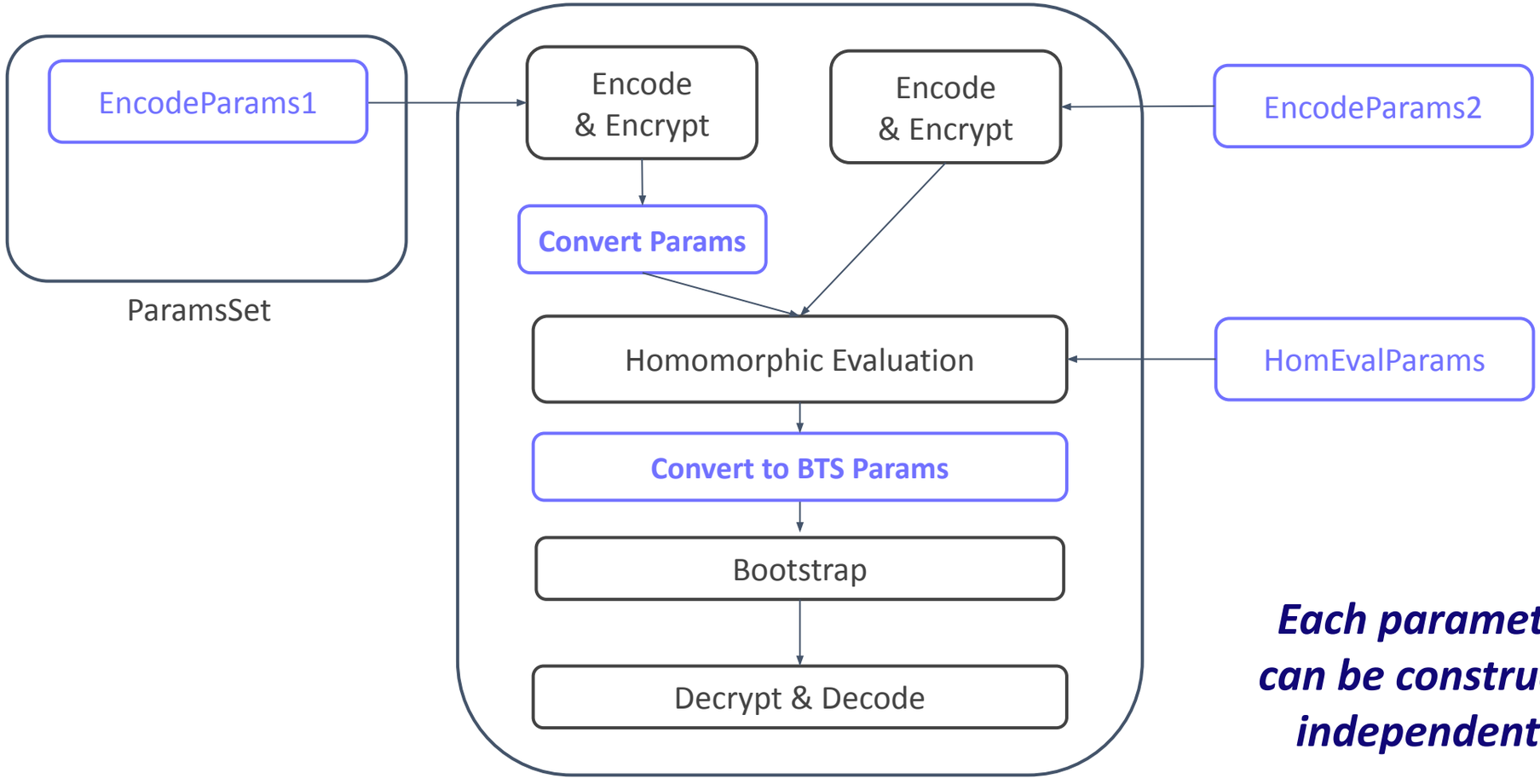
Choose best design between Performance and Abstraction!

Design Challenges

Design Challenges 1 : Evict Centralized Parameters



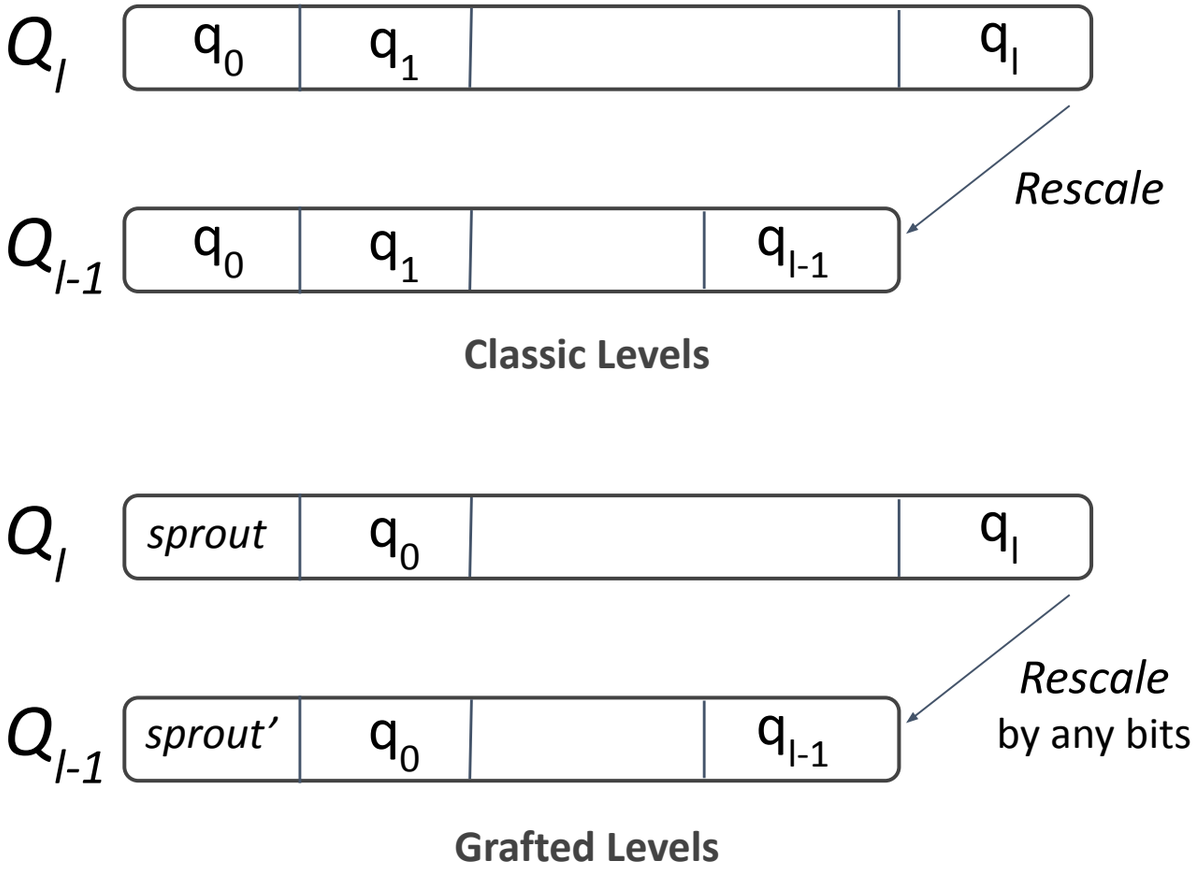
Design Challenges 1 : Evict Centralized Parameters



Advanced user experience

Each parameters can be constructed independently.

Design Challenges 2 : Redefine Levels



Hidden Limitation of Classic Levels

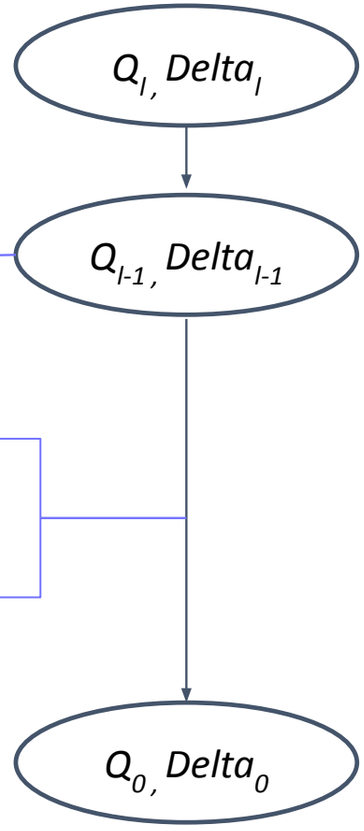
- Moduli chain is constructed in a single sequence.
- Moduli in the chain divides the next one.
- RNS system is truncated in a fixed order.
- One rescaling removes 1 word.
- Rescaling amount is NTT-able.

*[CCS25] Grafting: Decoupled Scale Factors and Modulus in RNS-CKKS, Cheon et al.

Design Challenges 2 : Redefine Levels

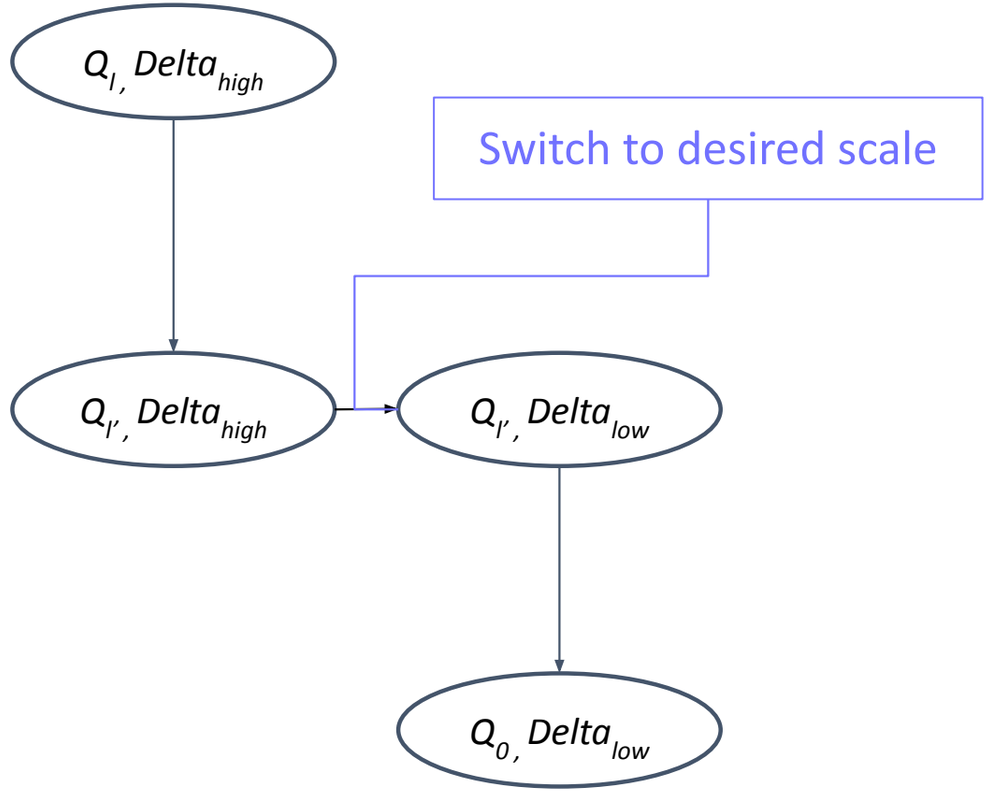
Stabilize Delta by choosing proper Q_l

Follows recurrence relation
$Delta_i = Delta_{i+1}^2 / (Q_{l+1} / Q_i)$



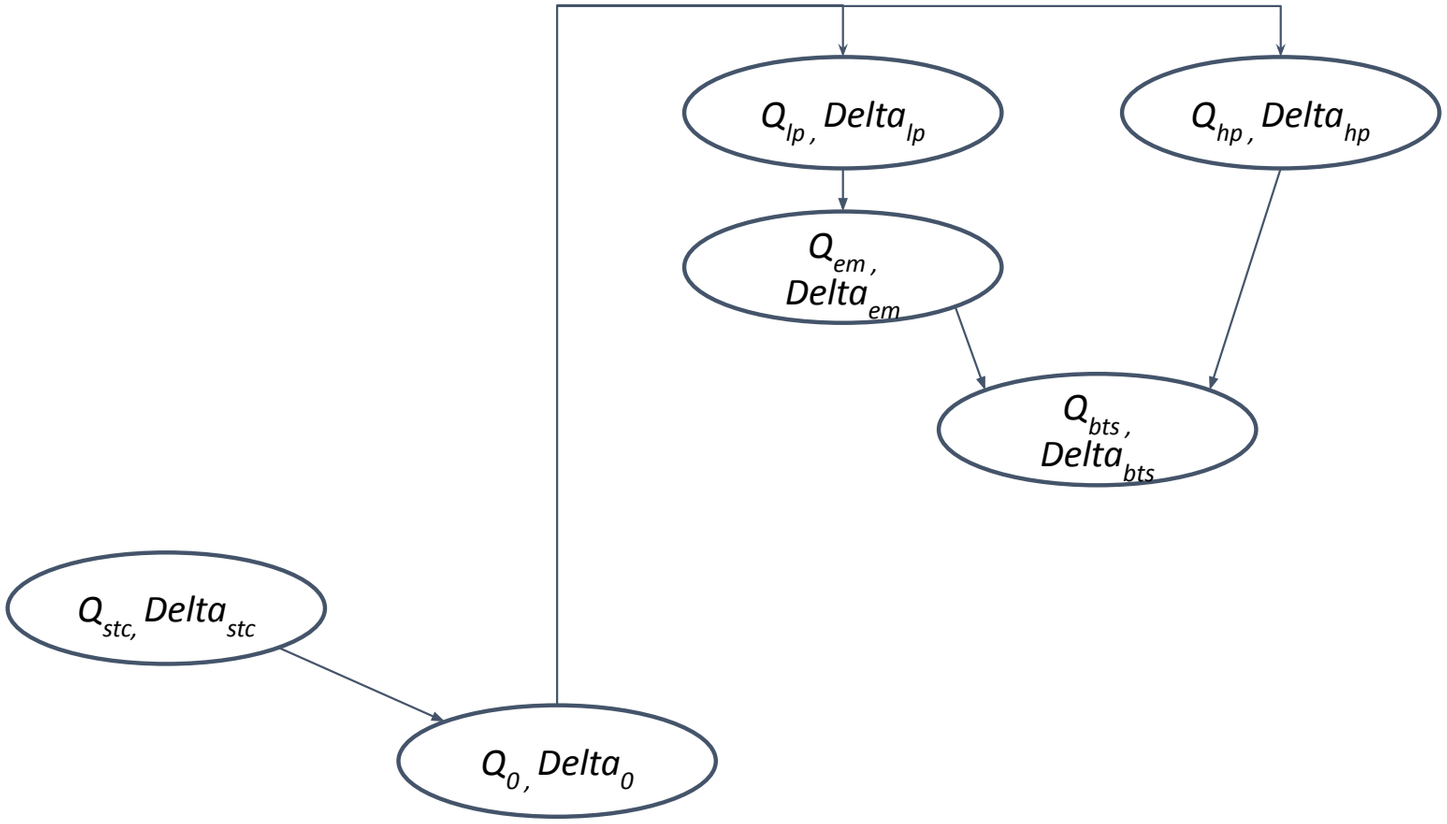
A simple HEaaN2 Levels

Design Challenges 2 : Redefine Levels



A modulus-optimized Levels

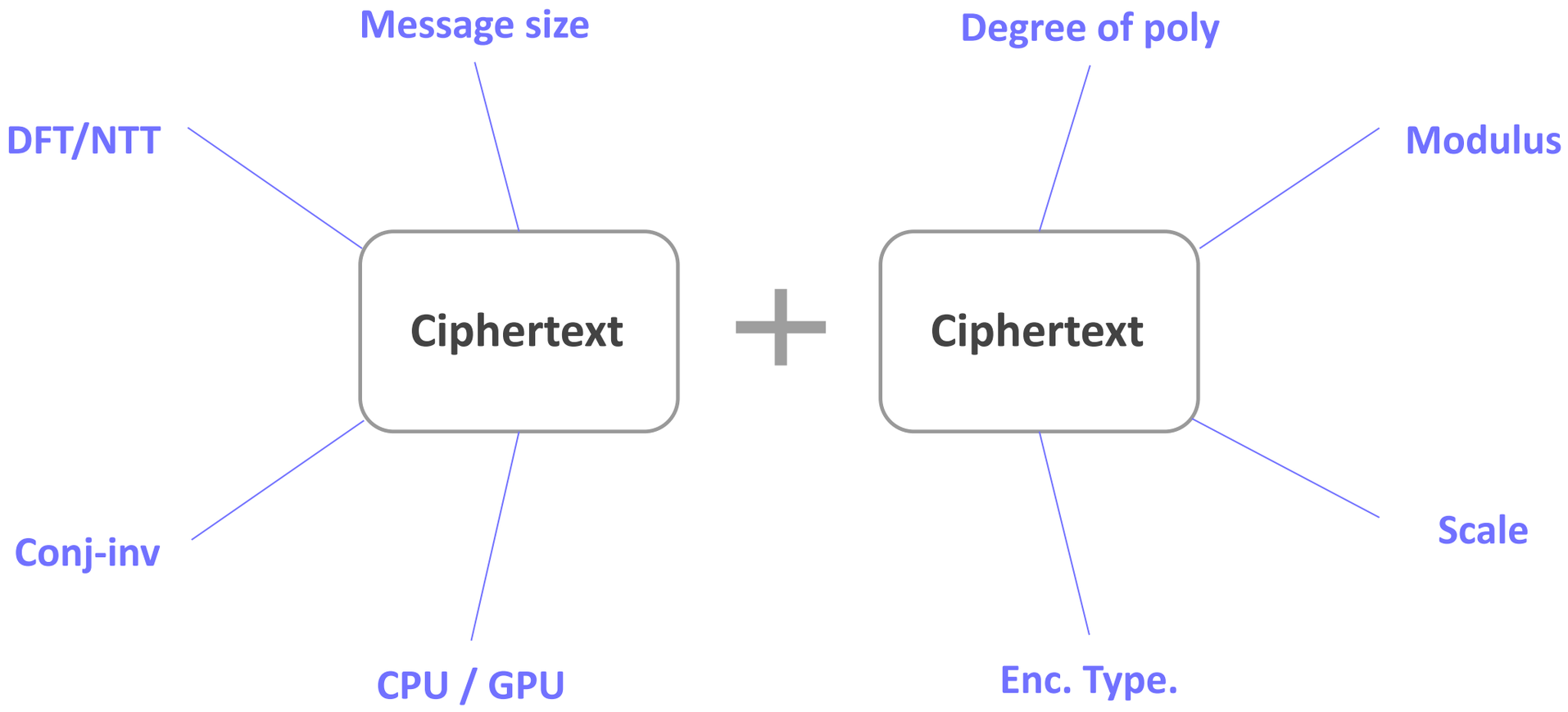
Design Challenges 2 : Redefine Levels



Example of advanced use of Levels*

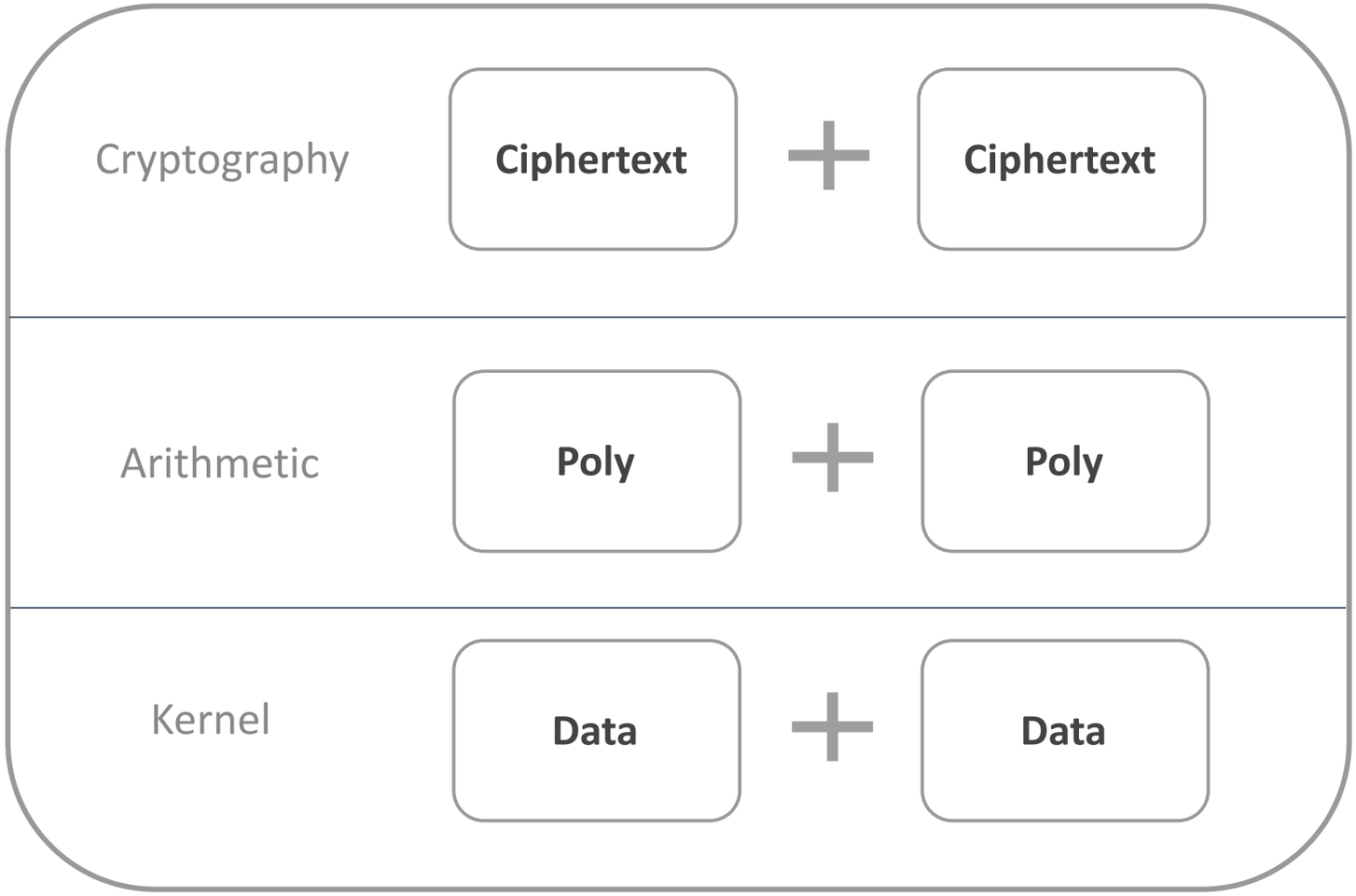
*[CCS25] Leveraging Discrete CKKS to Bootstrap in High Precision, Choe et al.

Design Challenges 3 : Allow vast design space



Distribute complexity through layers

Design Challenges 3 : Allow vast design space



Encoding
size of message
scale
dft

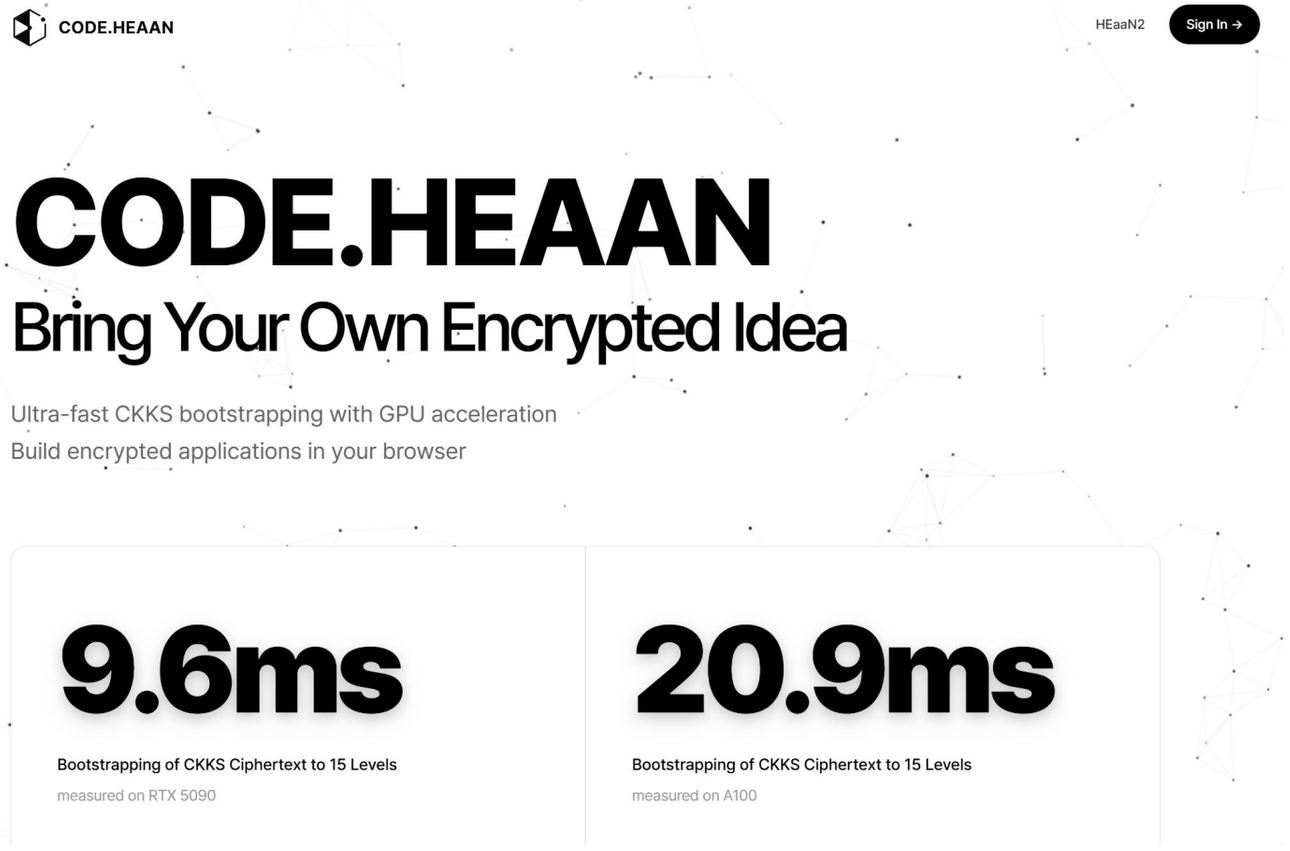
Encryption
power
rank
num secrets

PolyRing
degree
modulus
NTT

Device

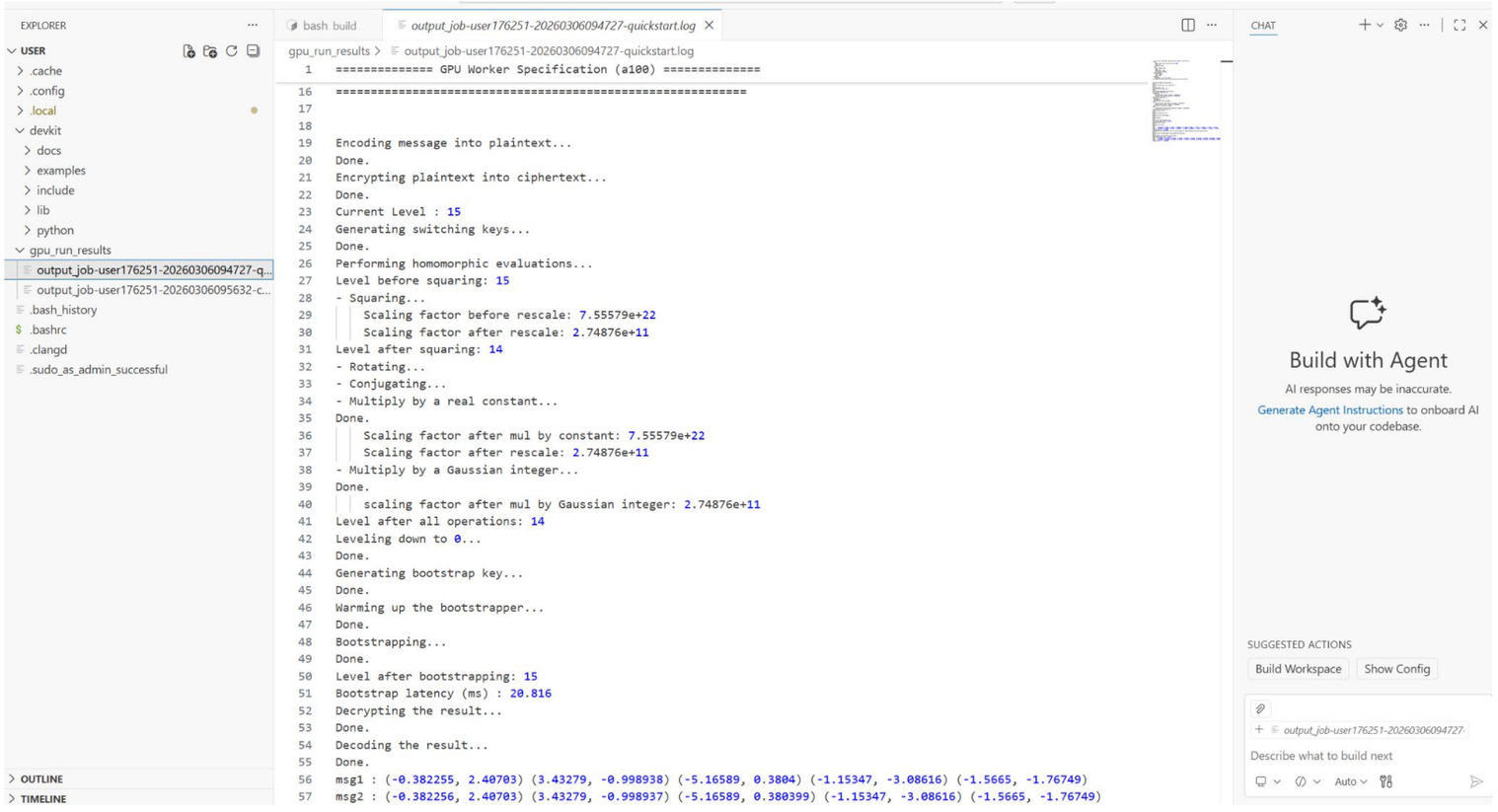
Conclusion

Conclusion



- ***CKKS only.***
- ***Closed-source.***
- ***SOTA performance.***
- ***Available at*** : heaan.io

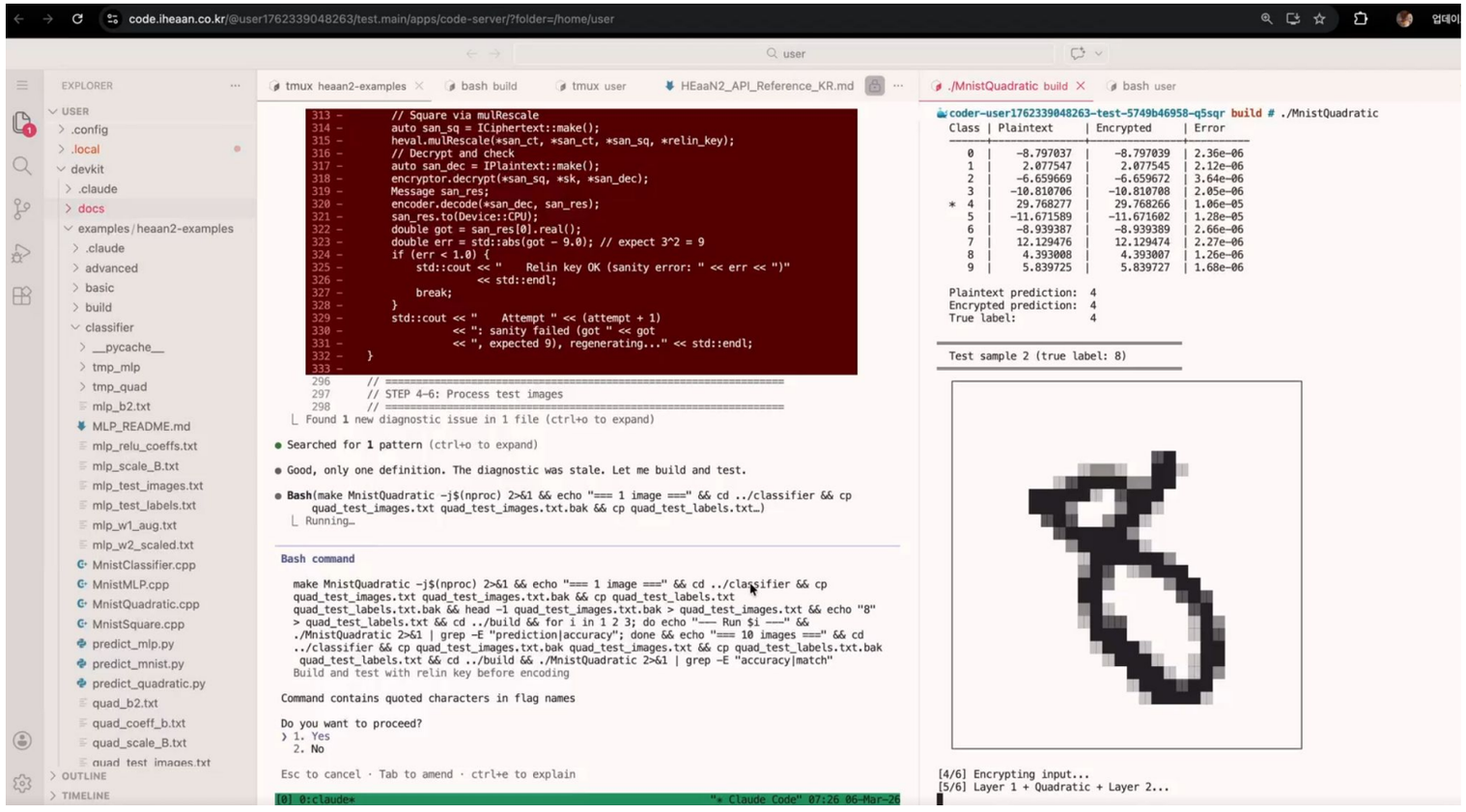
Conclusion



- ***CKKS only.***
- ***Closed-source.***
- ***SOTA performance.***
- ***Available at*** : heaan.io

Again, what is HEaaN2

Conclusion



- *CKKS only.*
- *Closed-source.*
- *SOTA performance.*
- *Available at* : heaan.io

Thank you for listening.

CRYPTO LAB
주식회사 크립토랩