Samuel TAP

10.10.2023

# Fully Homomorphic Encryption I

## Zama’s FHE compiler

ZAMA

# Agenda

# FHE

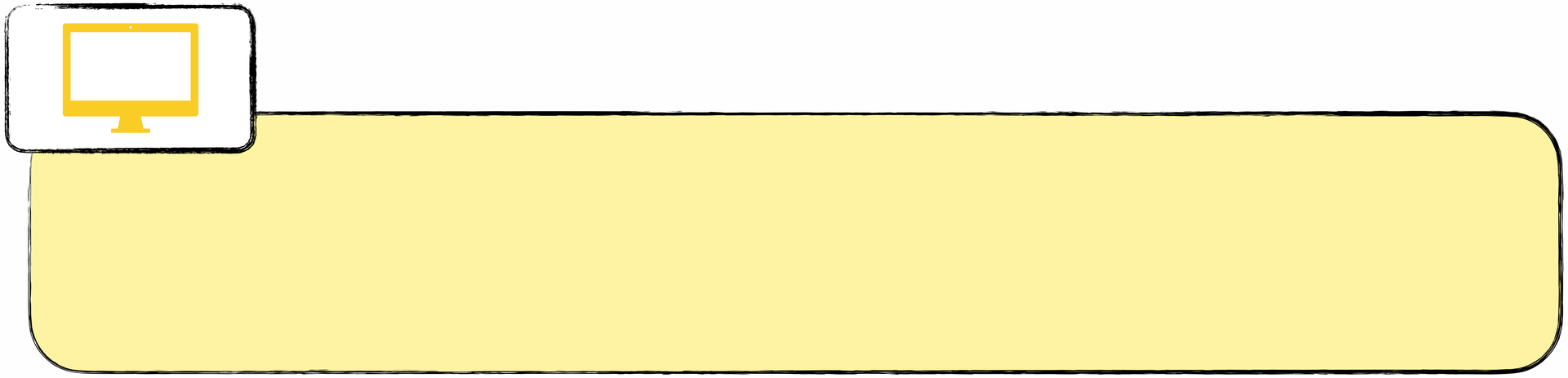
What is FHE ?

Past

# Cloud Computing

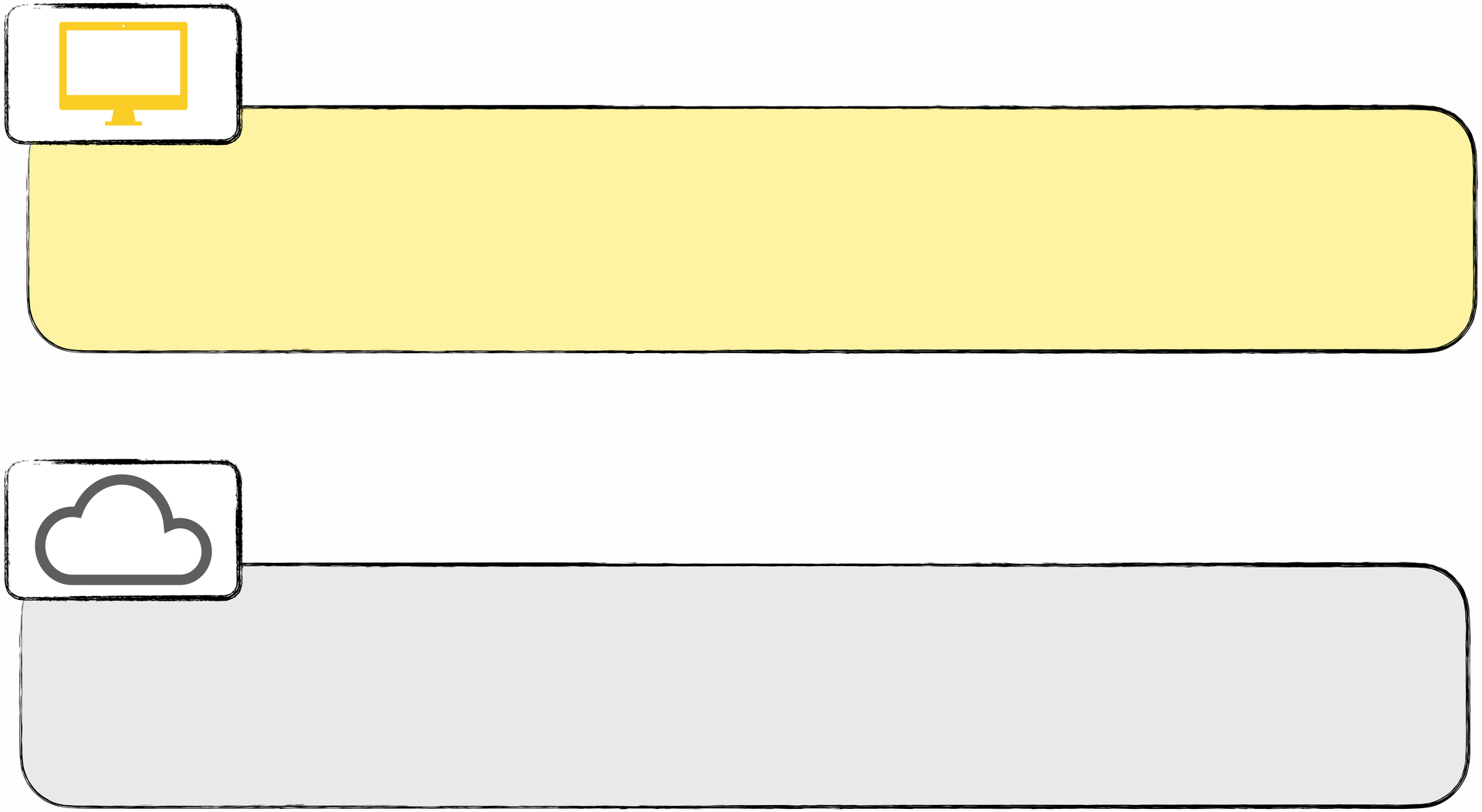
Past

# Cloud Computing



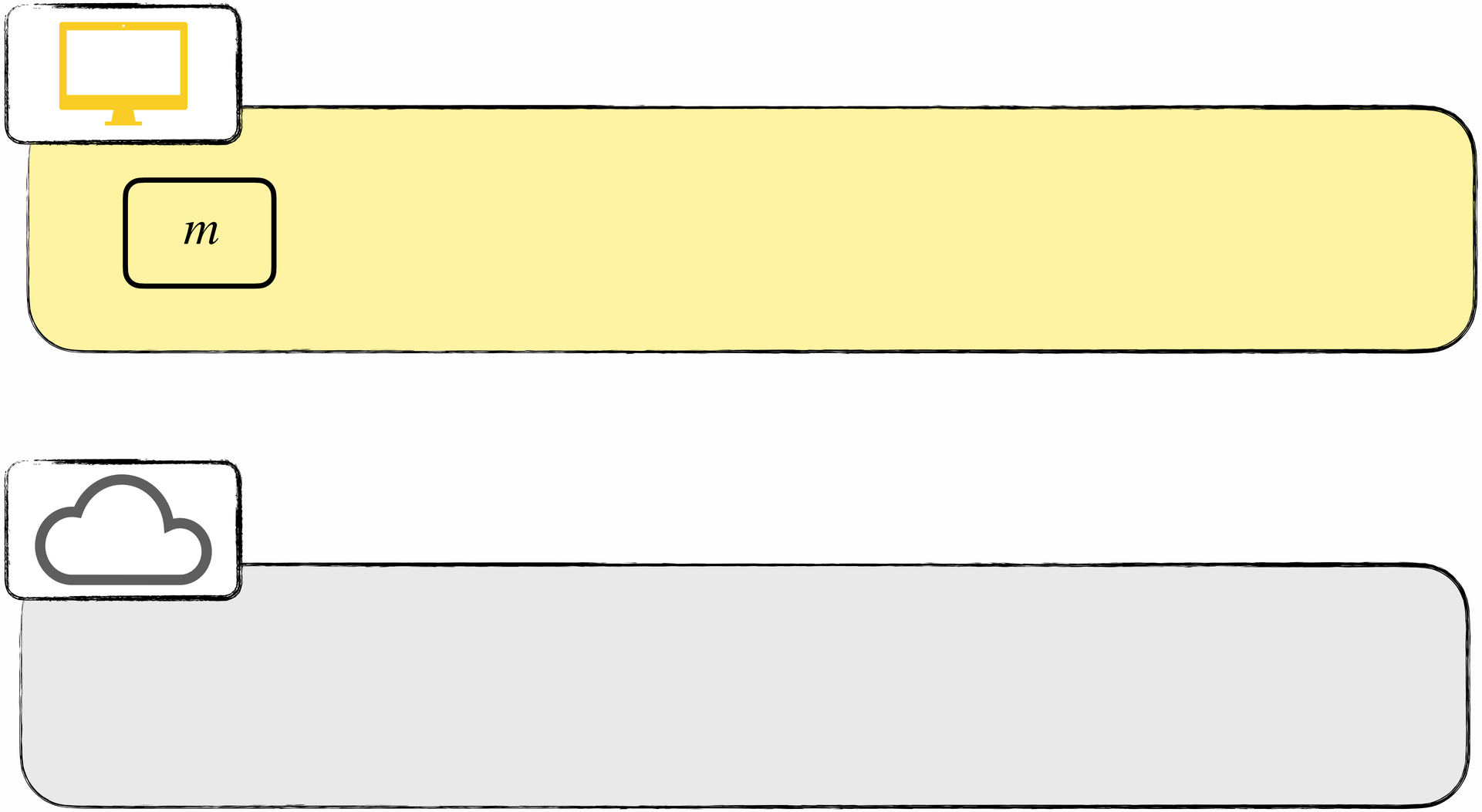
Past

# Cloud Computing



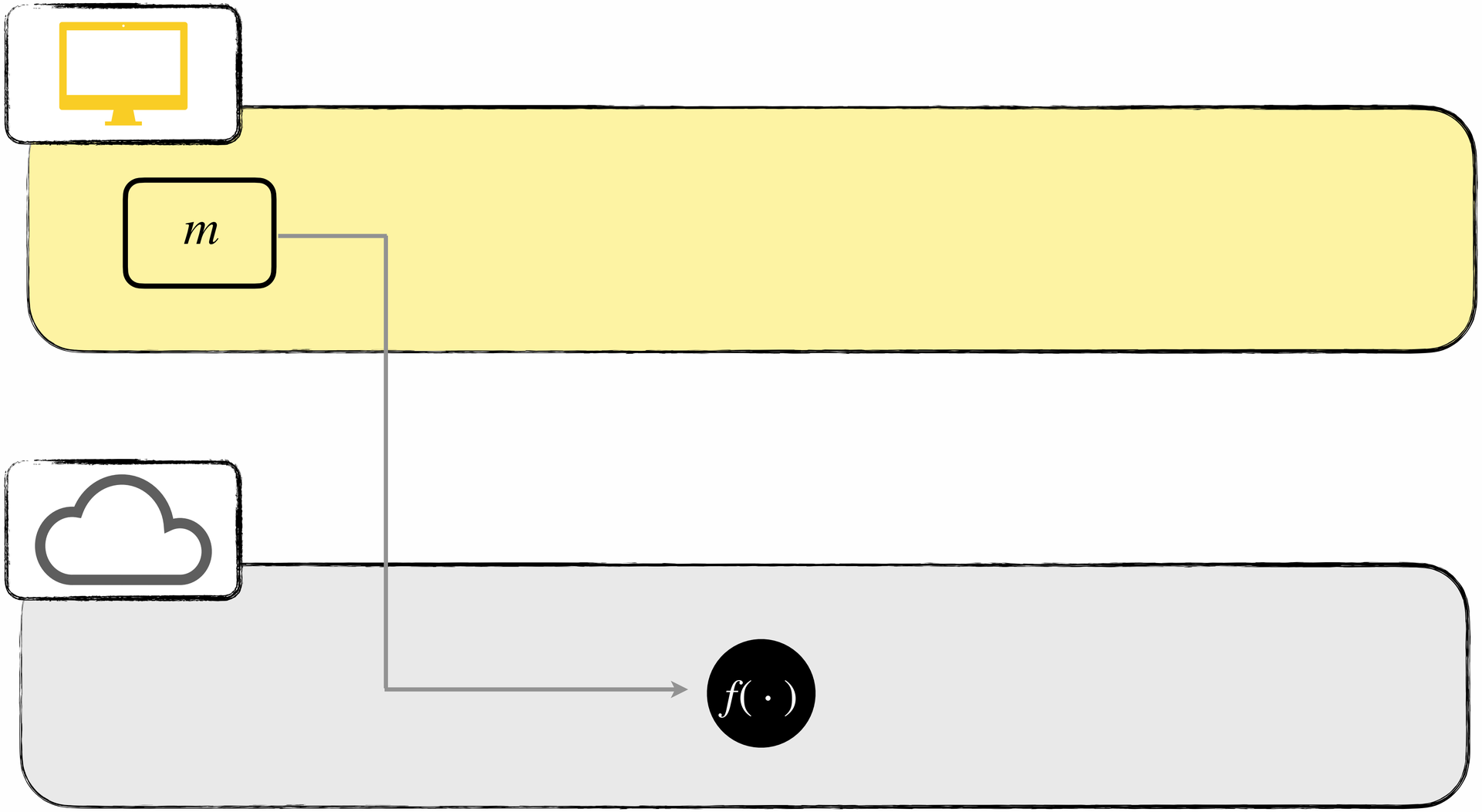
Past

# Cloud Computing



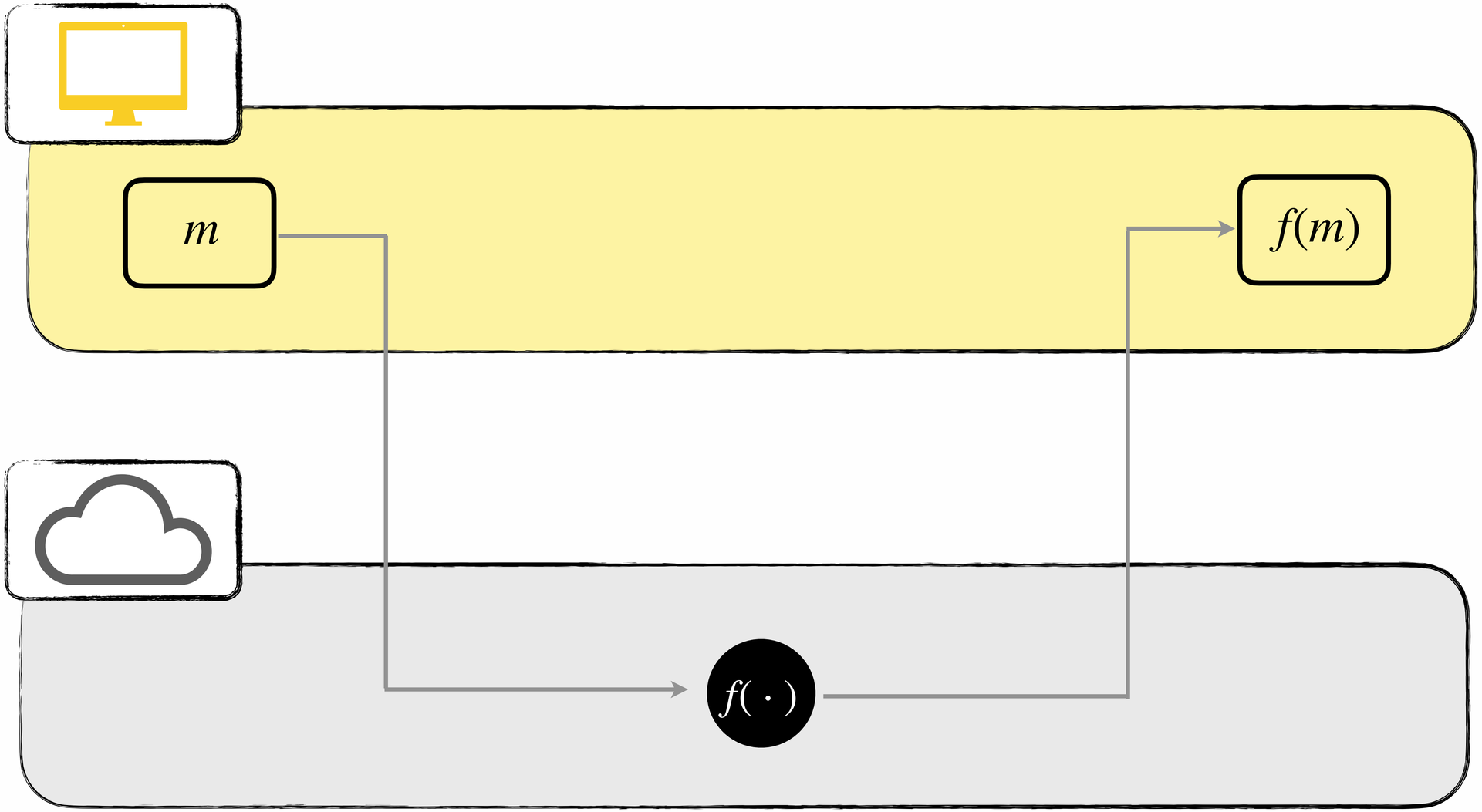
Past

# Cloud Computing



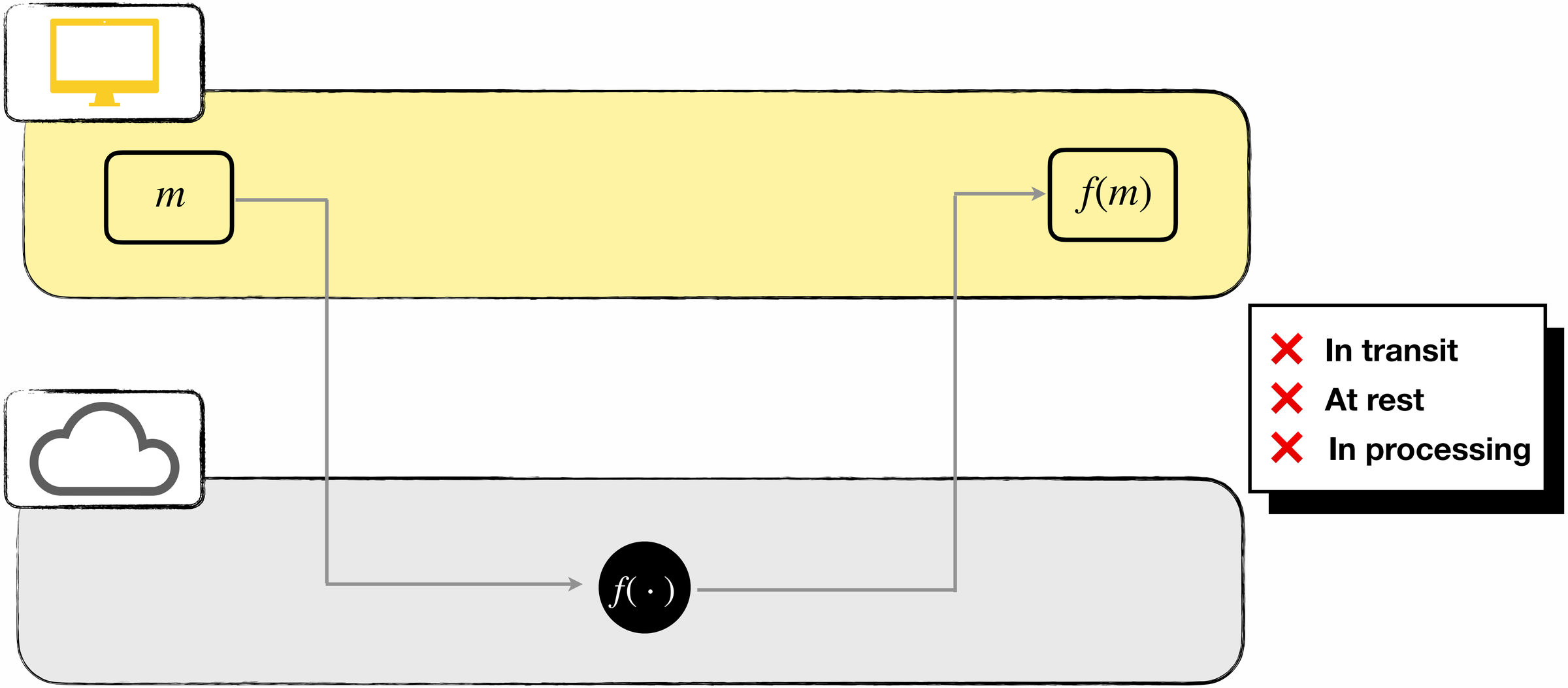
Past

# Cloud Computing



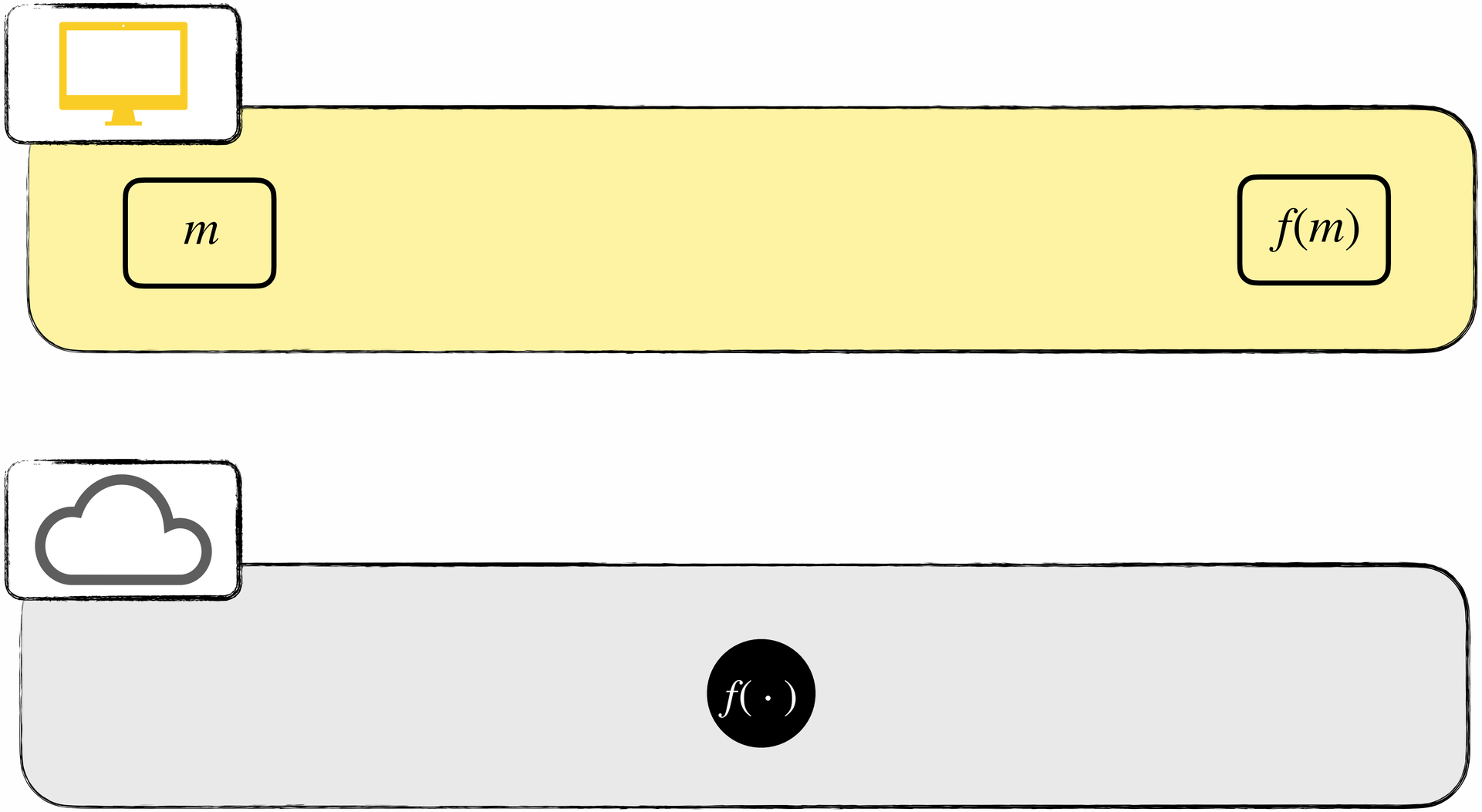
Past

# Cloud Computing



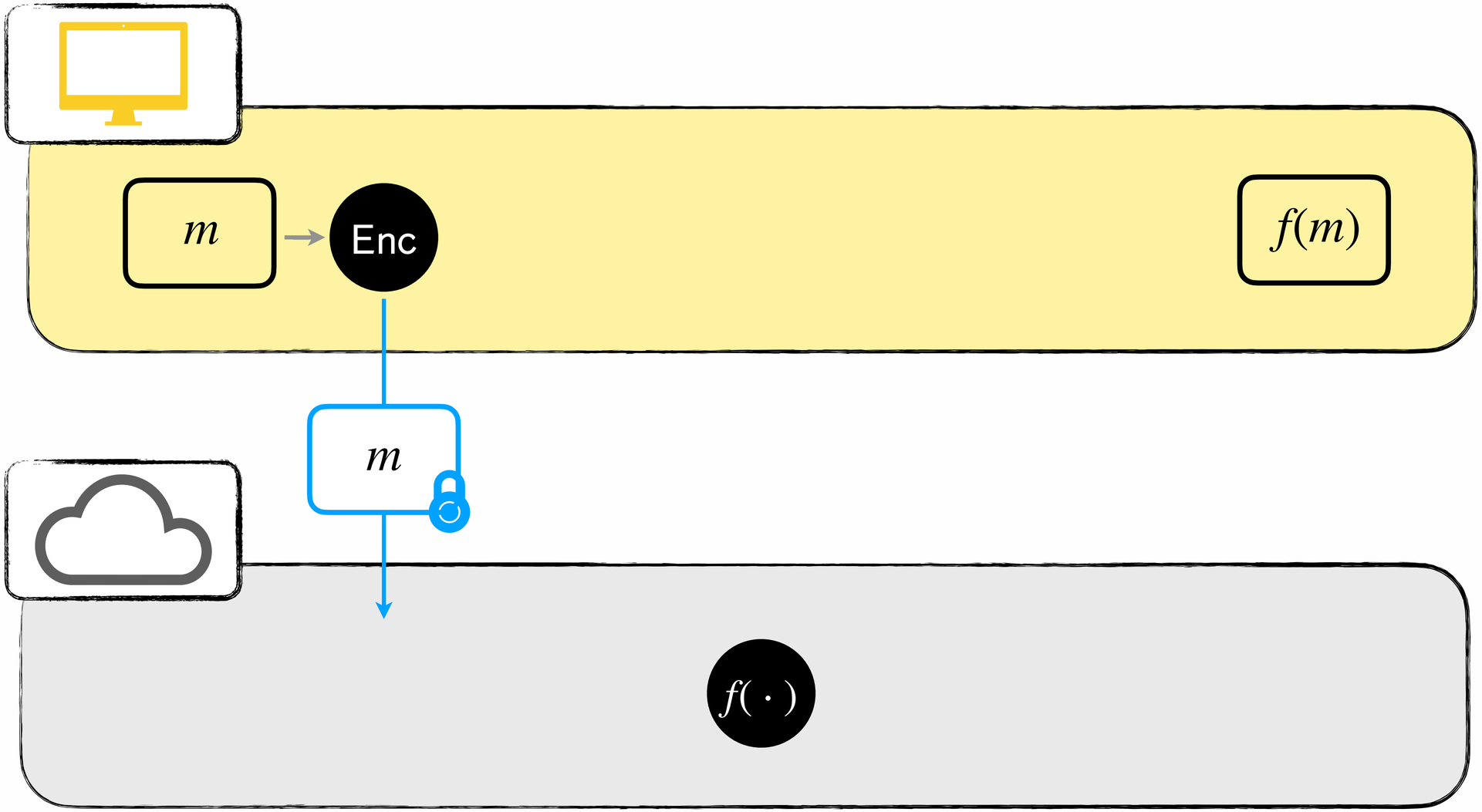
Past Present

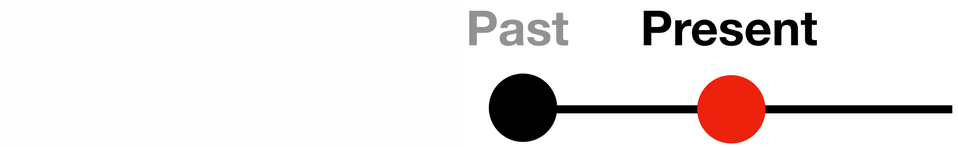
# Cloud Computing



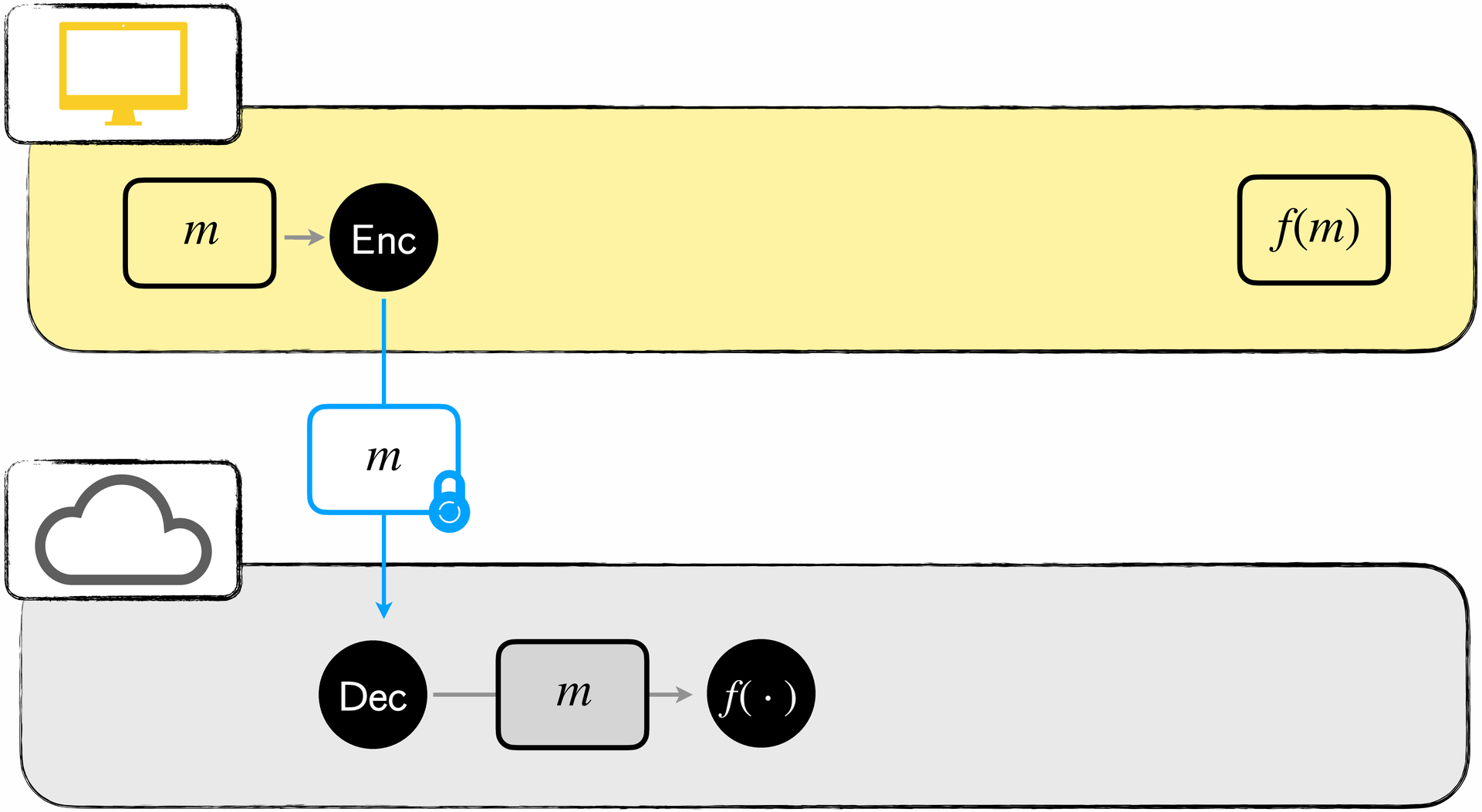
Past Present

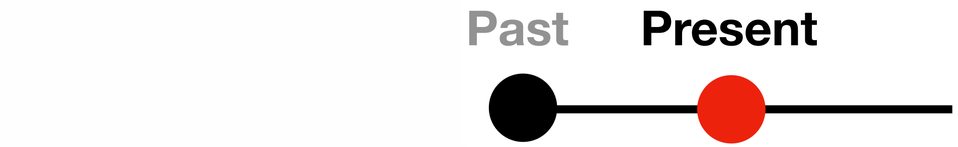
# Cloud Computing



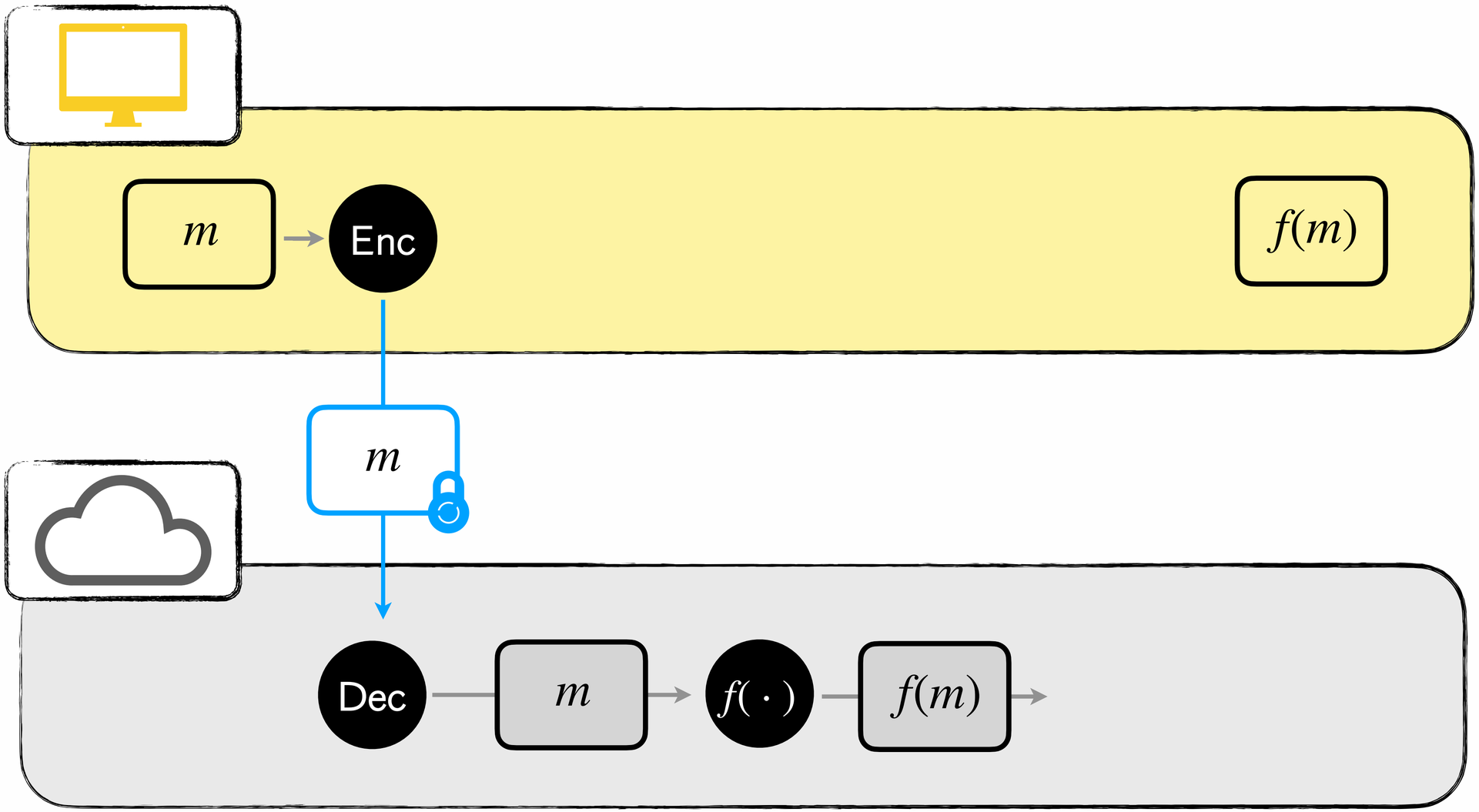



# Cloud Computing



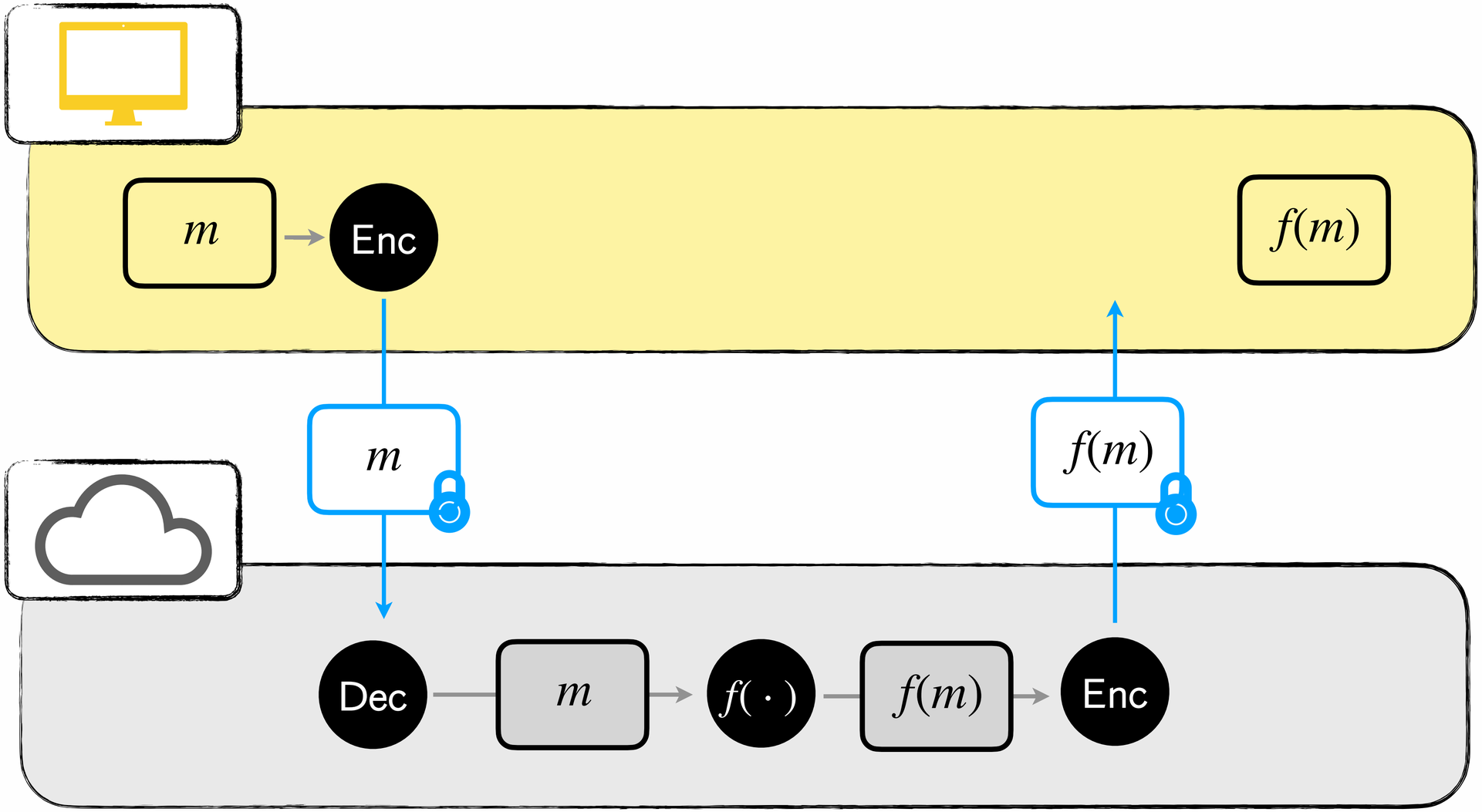



# Cloud Computing



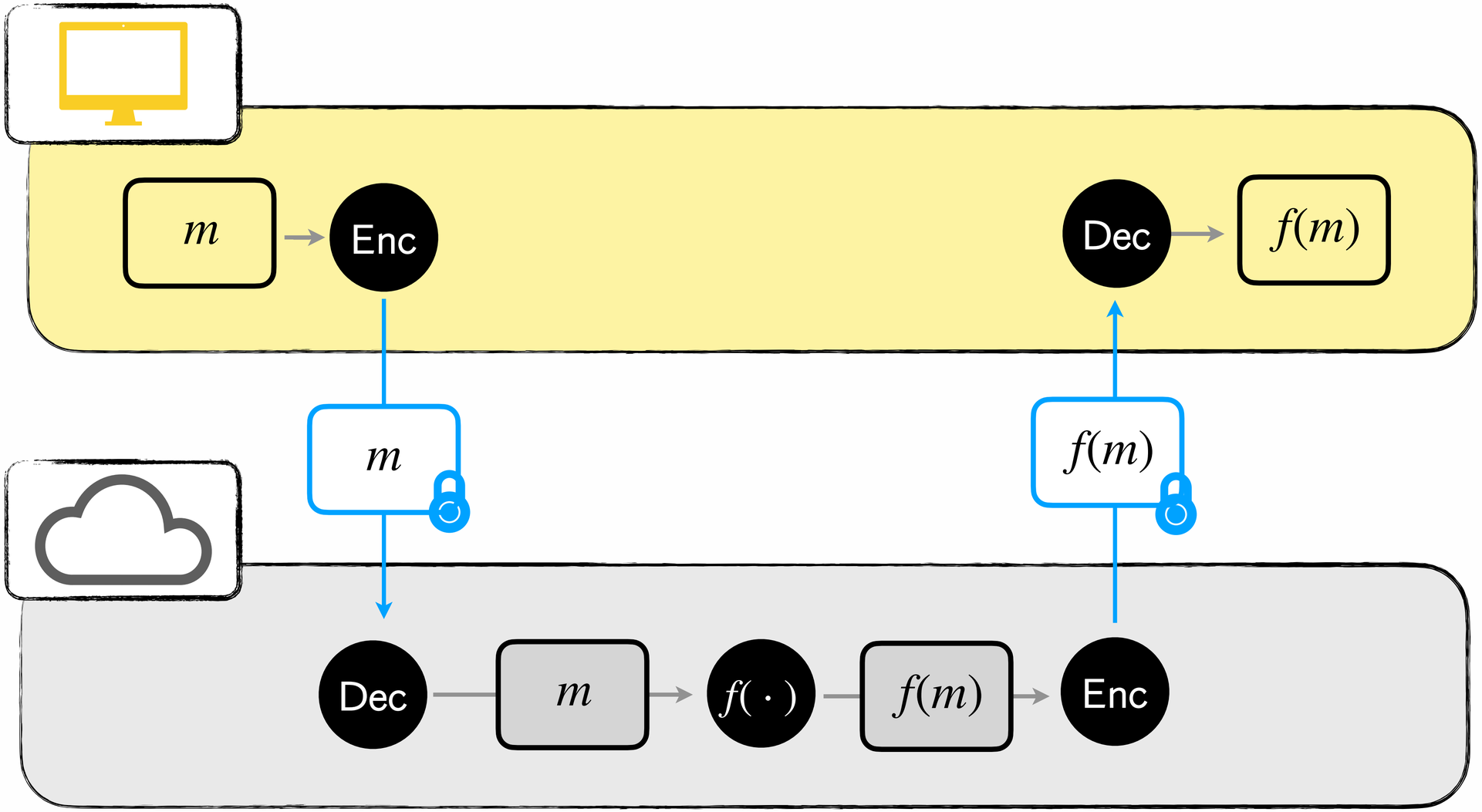
Past Present

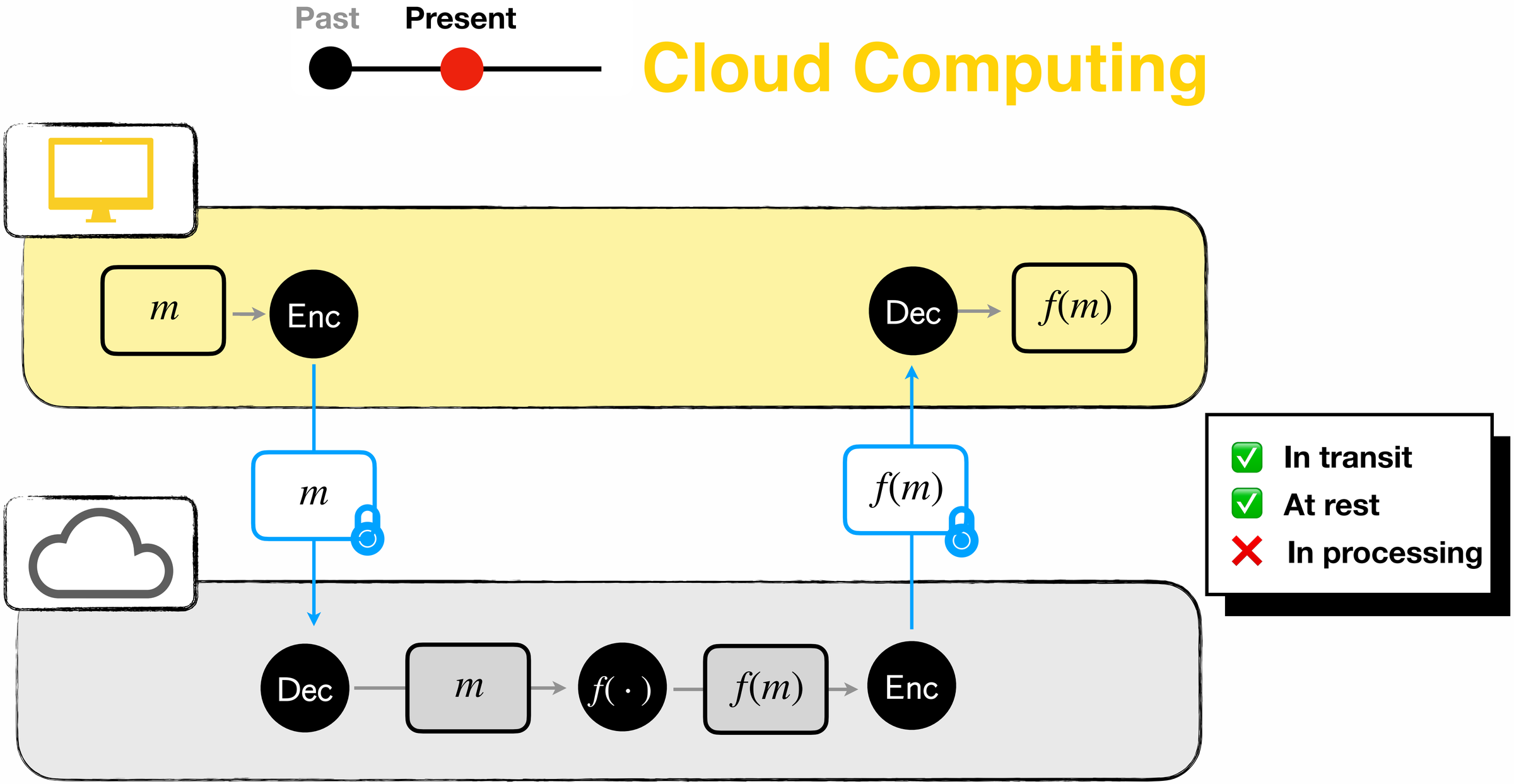
# Cloud Computing



Past Present

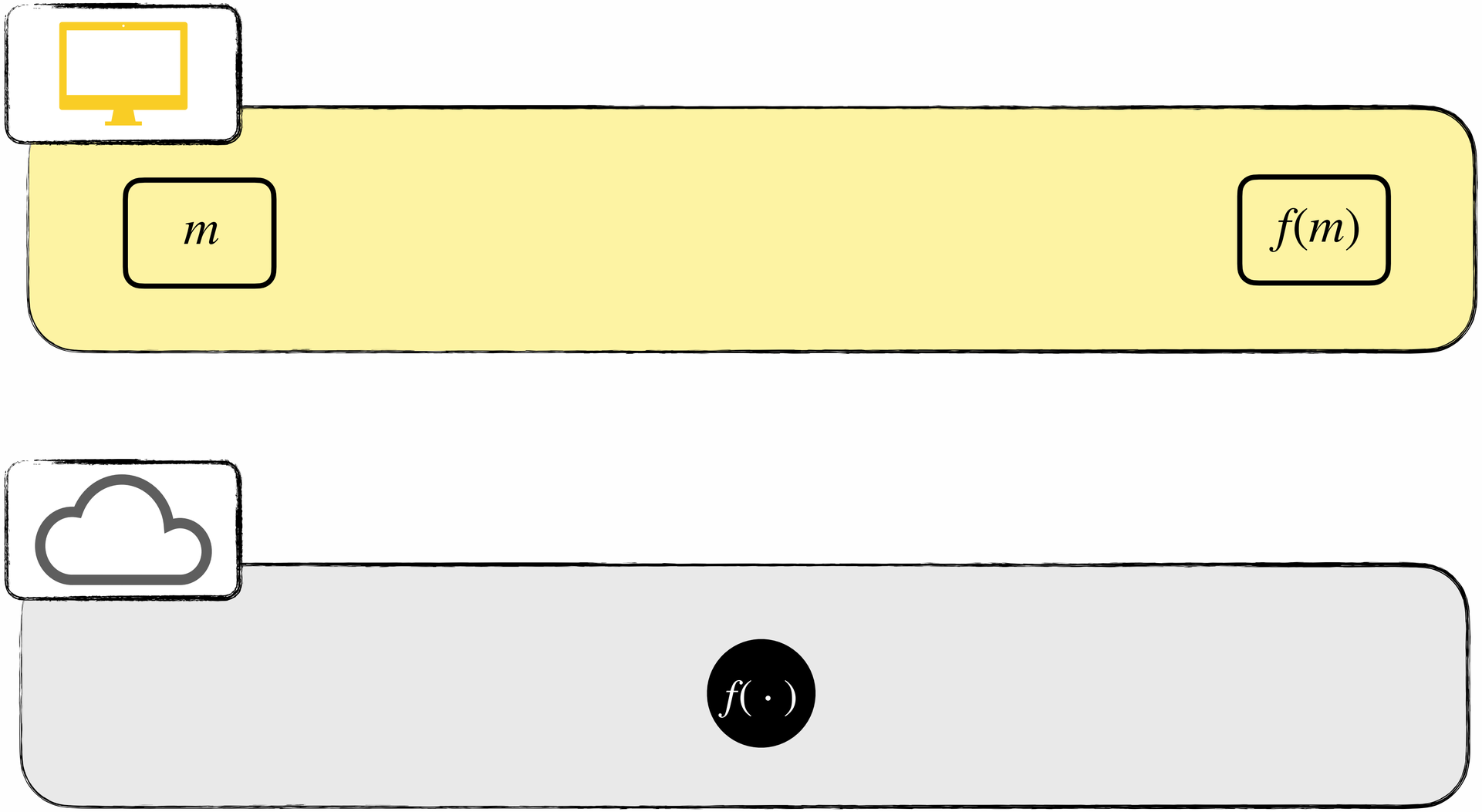
# Cloud Computing




Past
Present
Cloud Computing
m
Enc
Dec
f(m)
m
f(m)
Dec
m
f( · )
f(m)
Enc
In transit
At rest
In processing

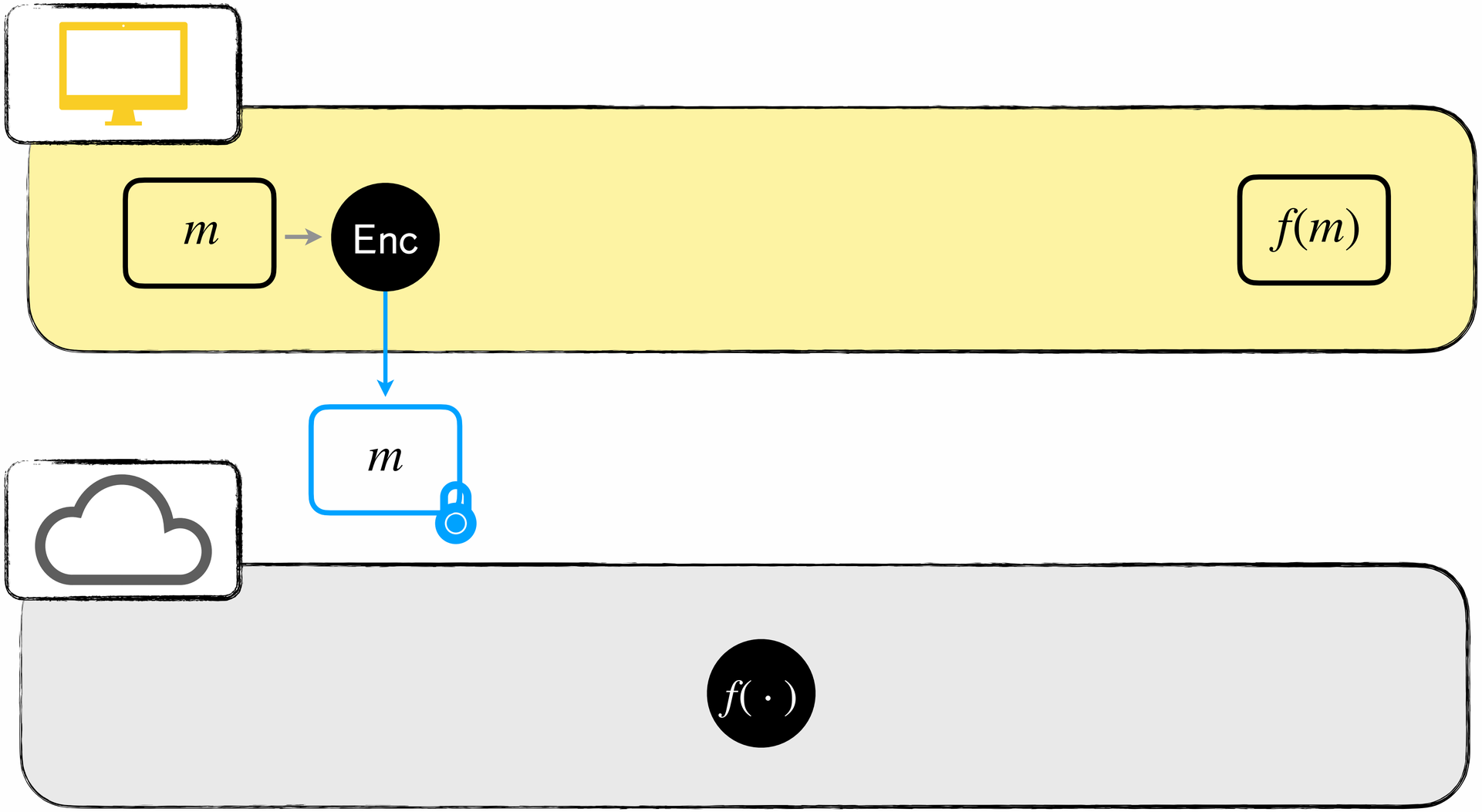
Past Present FHE

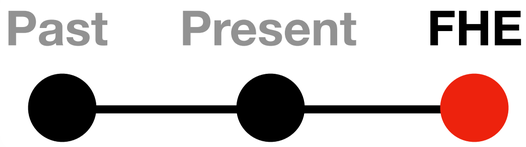
# Cloud Computing



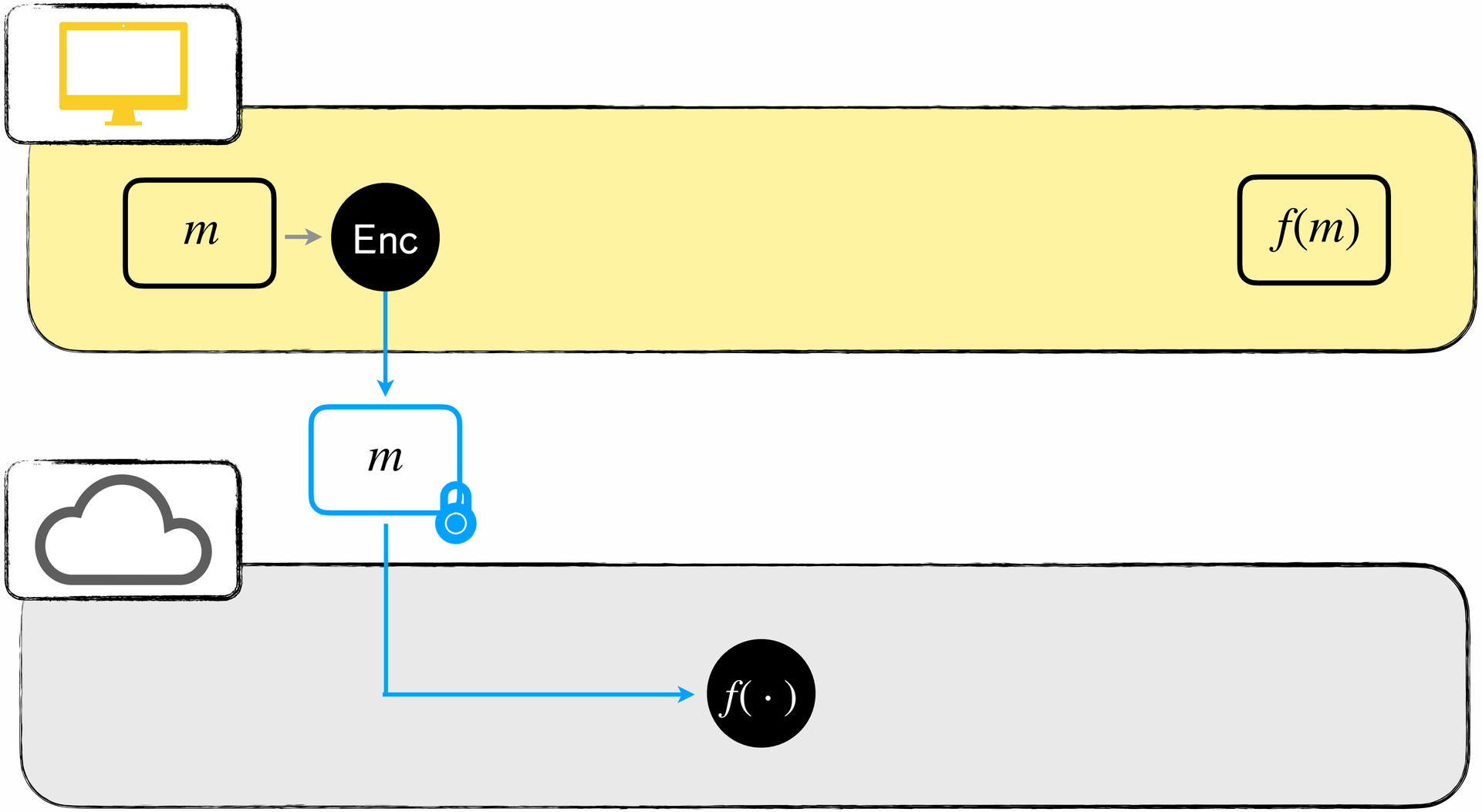
Past Present FHE

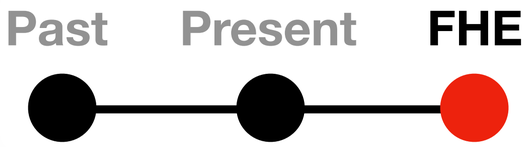
# Cloud Computing



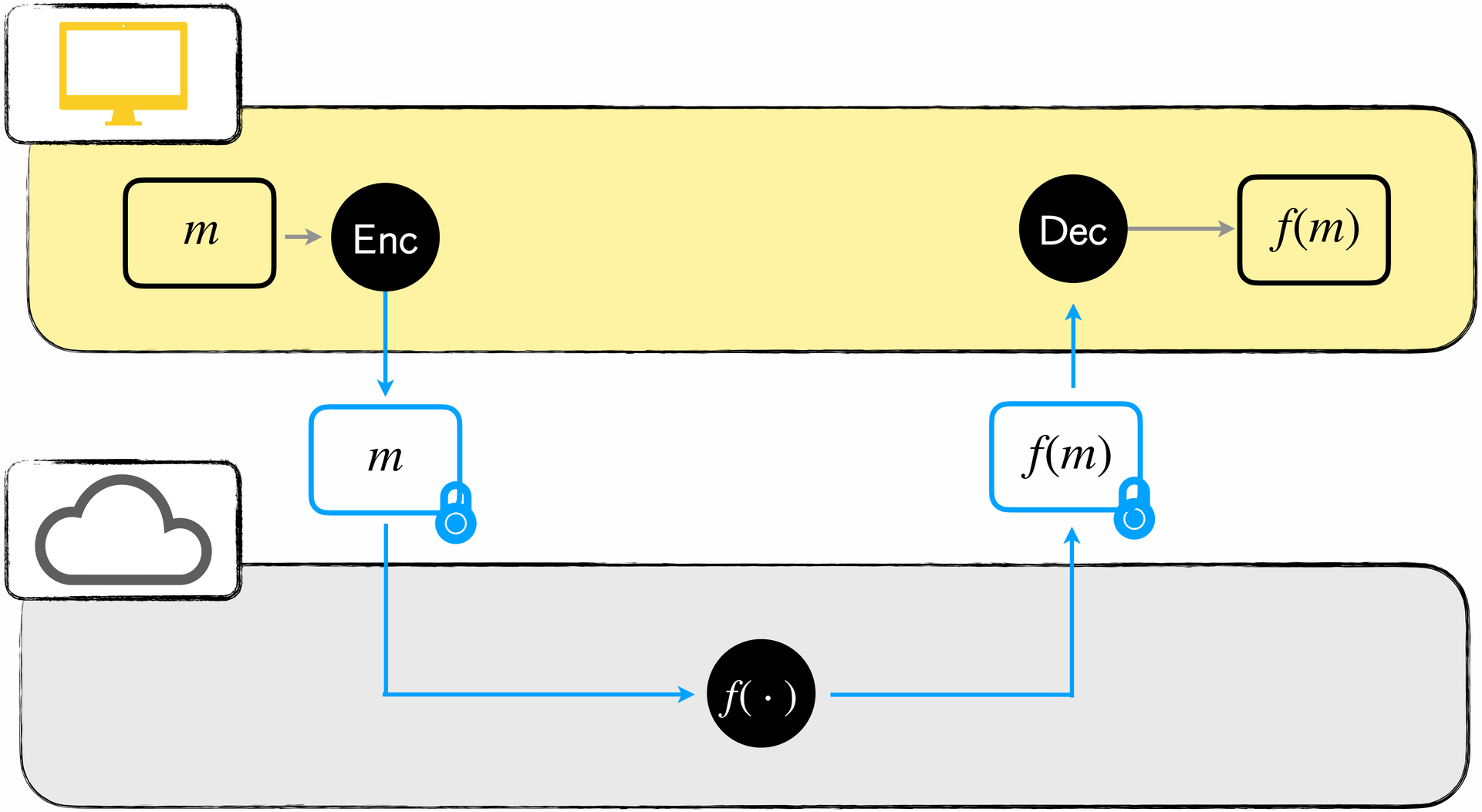


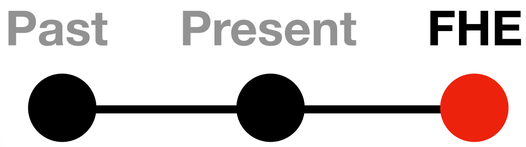

# Cloud Computing



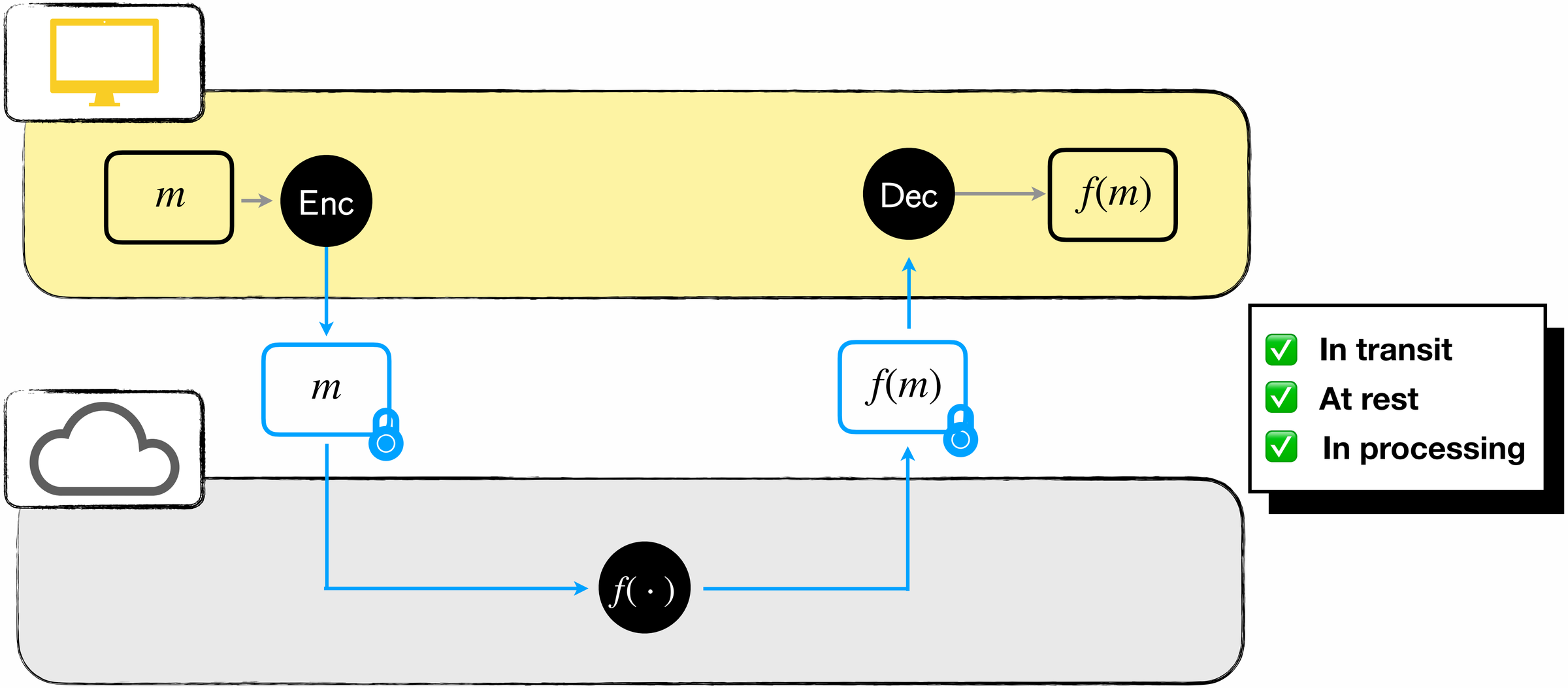



# Cloud Computing



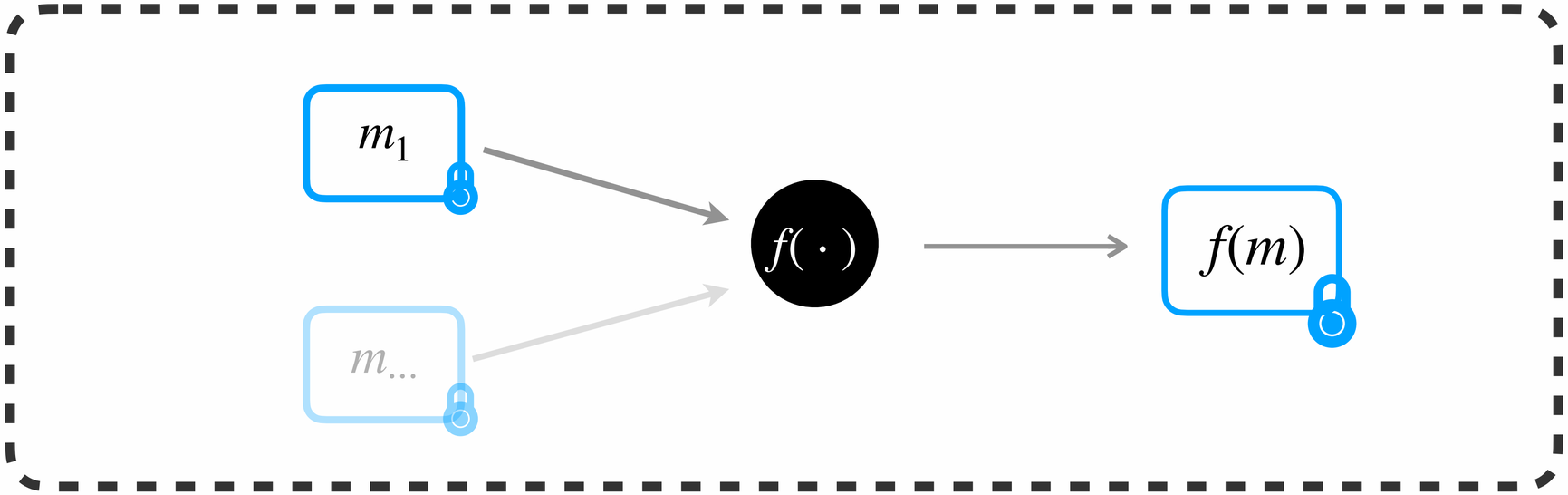



# Cloud Computing

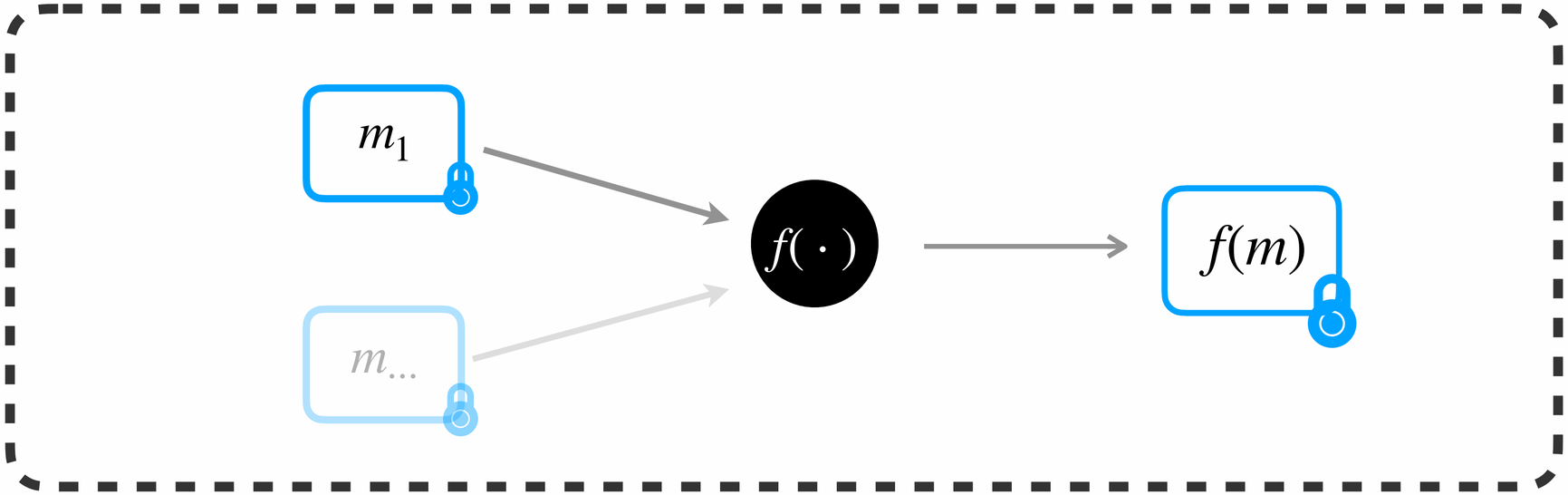



- ✅ In transit
- ✅ At rest
- ✅ In processing

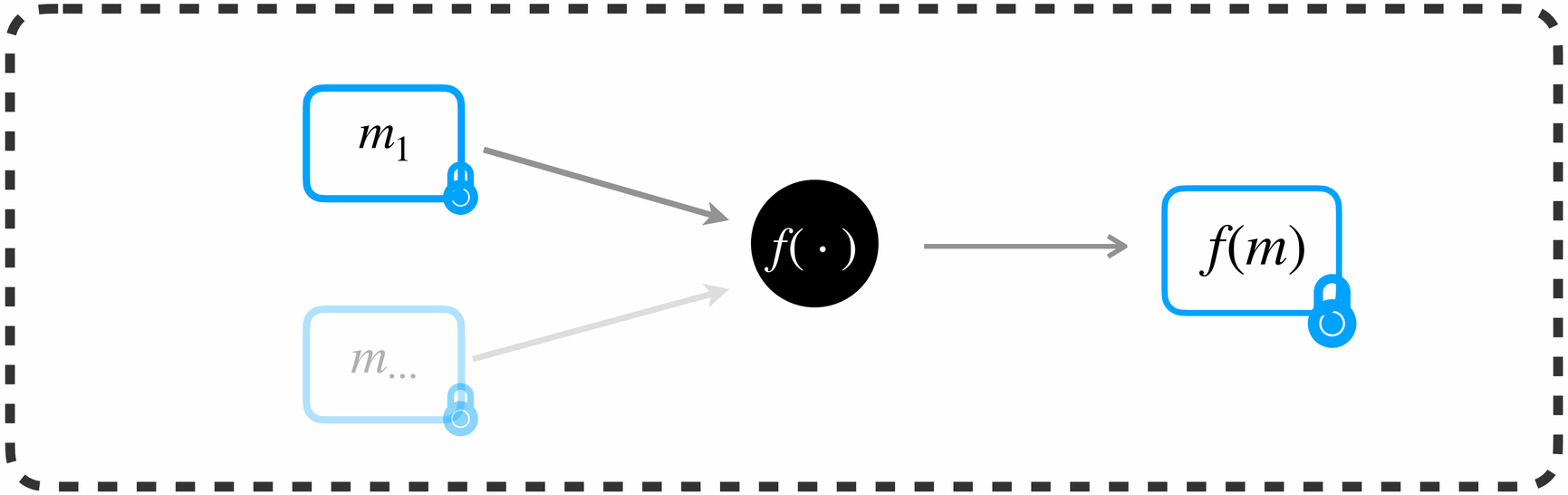
# FHE



# FHE

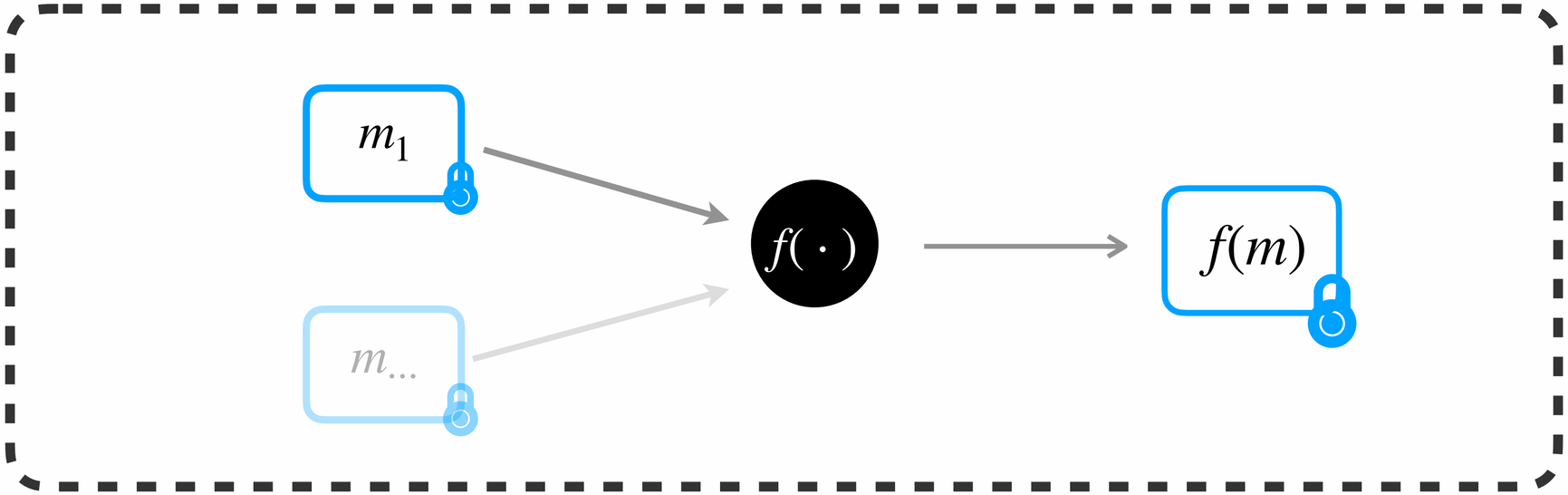


# FHE




$f$, a boolean circuit

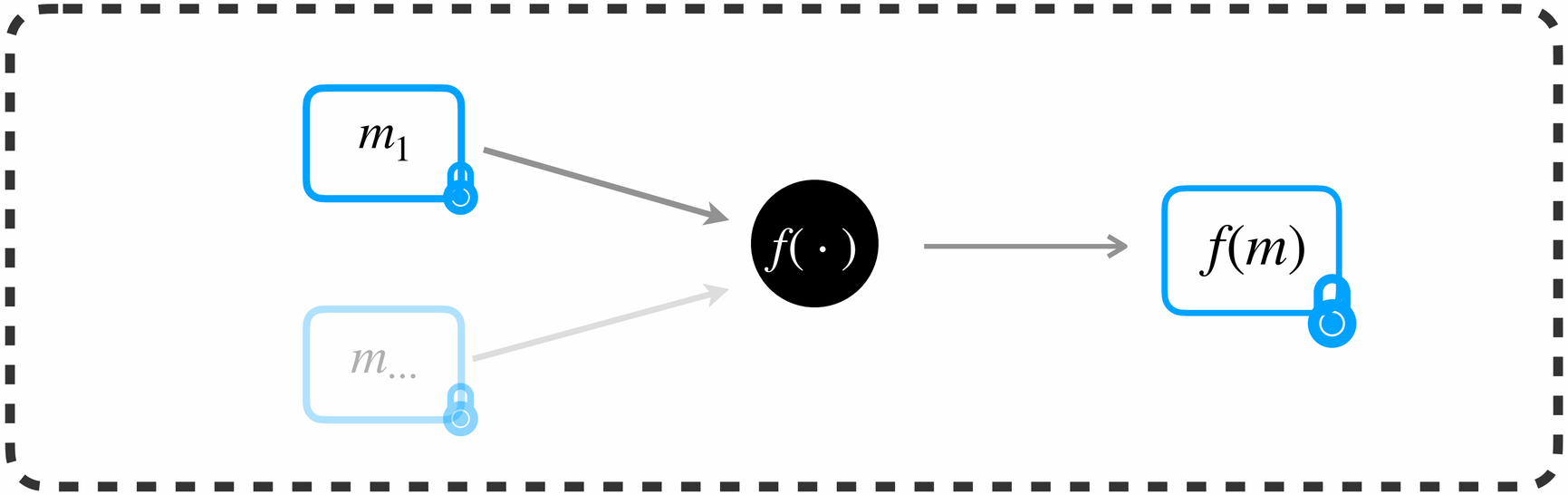
# FHE




$f$, a boolean circuit

$f$, a neural network

# FHE

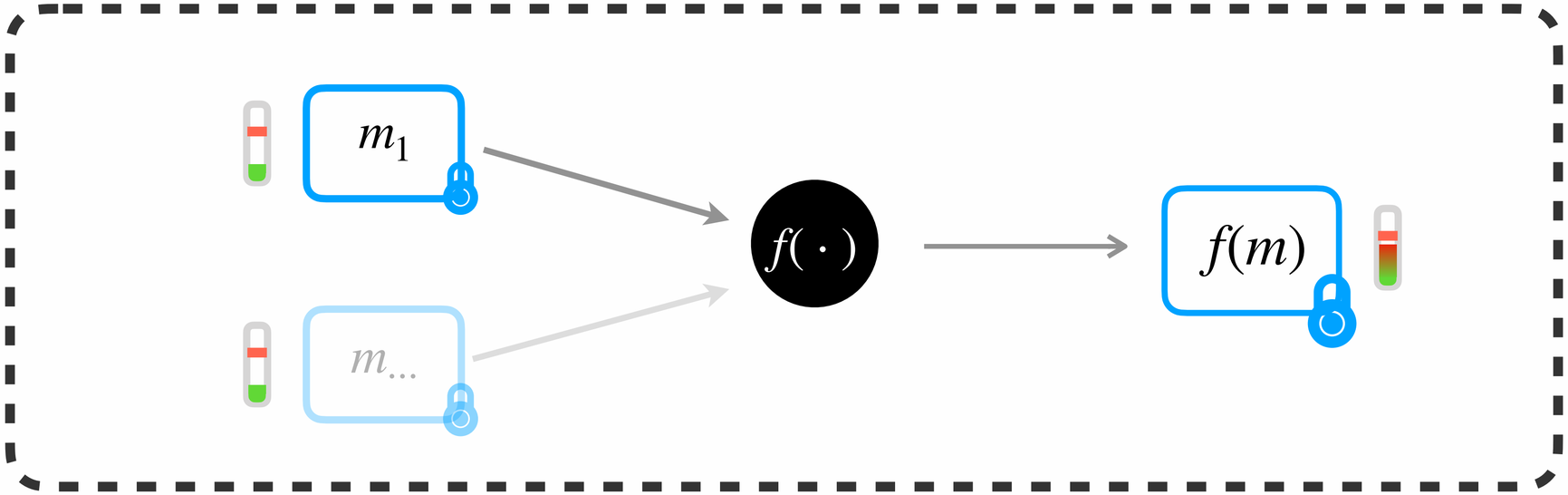



$f$, a boolean circuit

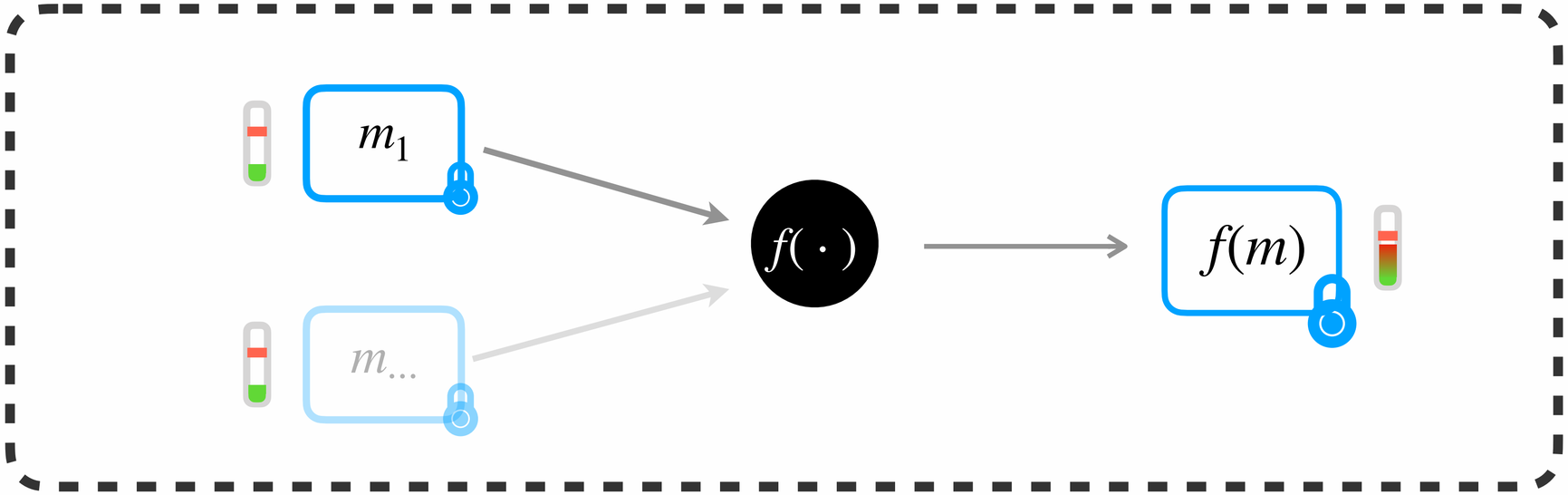
$f$, a neural network

$f$, any function

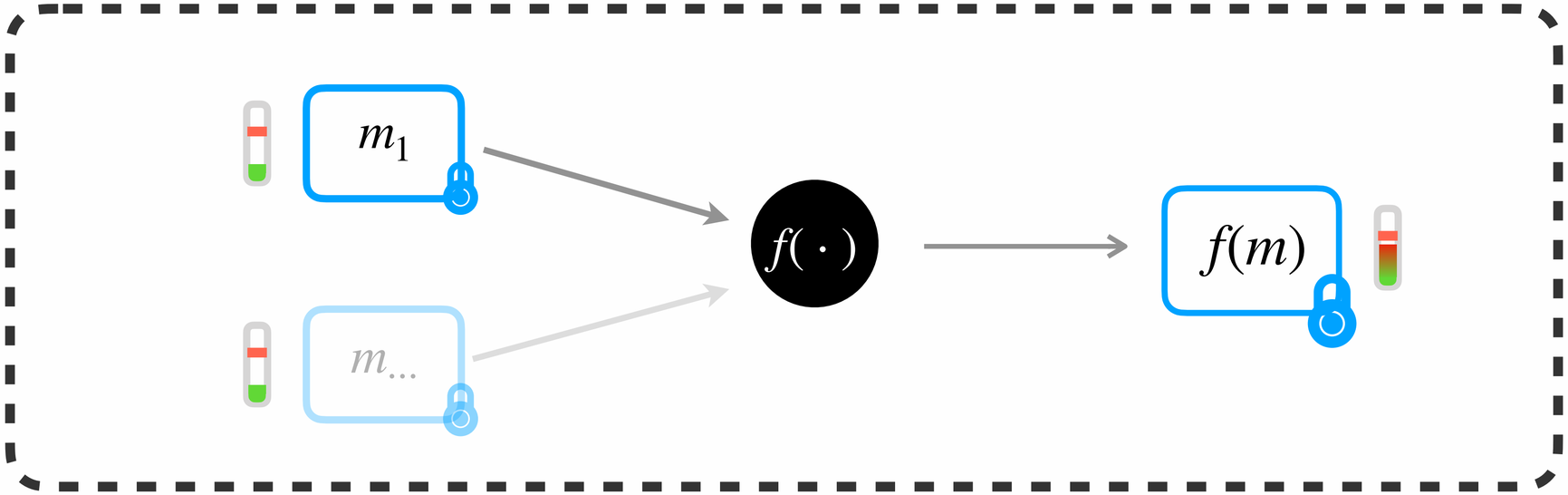
# FHE



# FHE

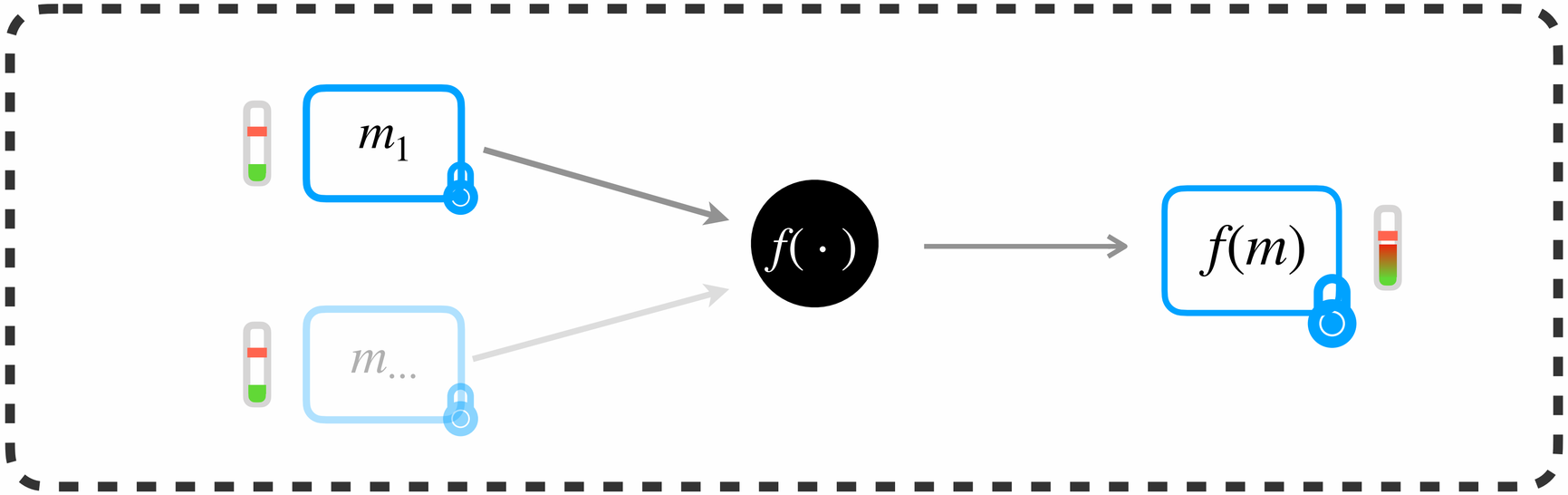


# FHE




**Most operations increase the noise**

# FHE



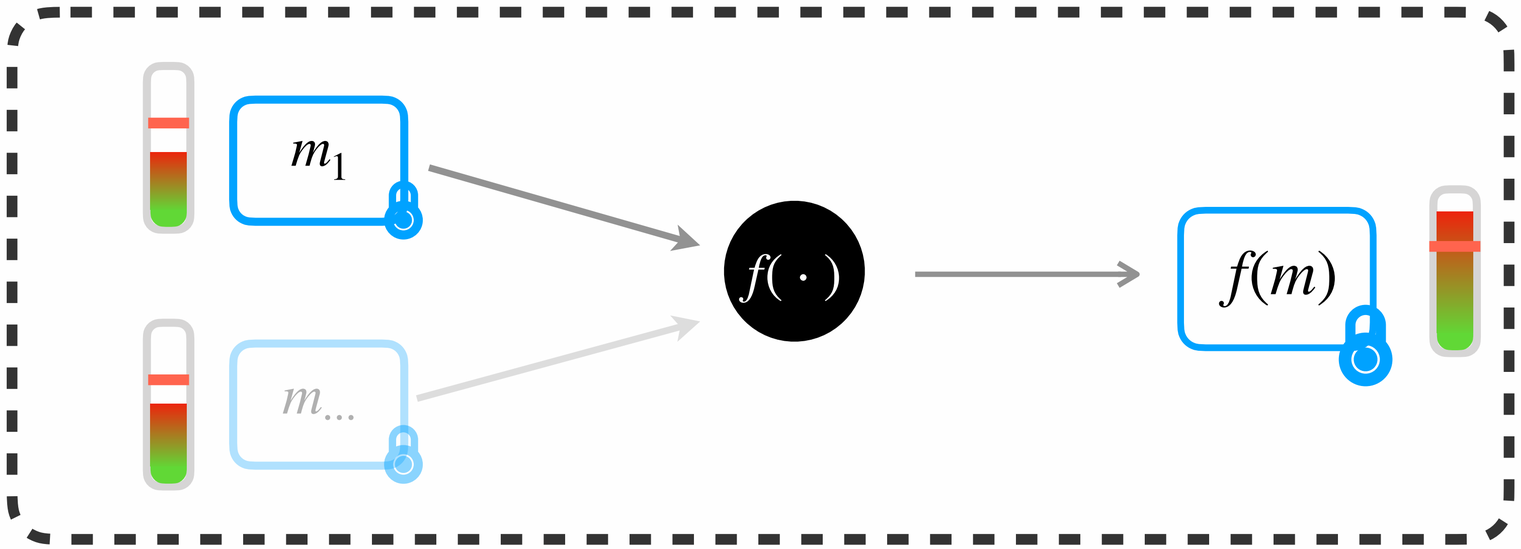

Most operations increase the noise

Too much noise leads to incorrect decryption

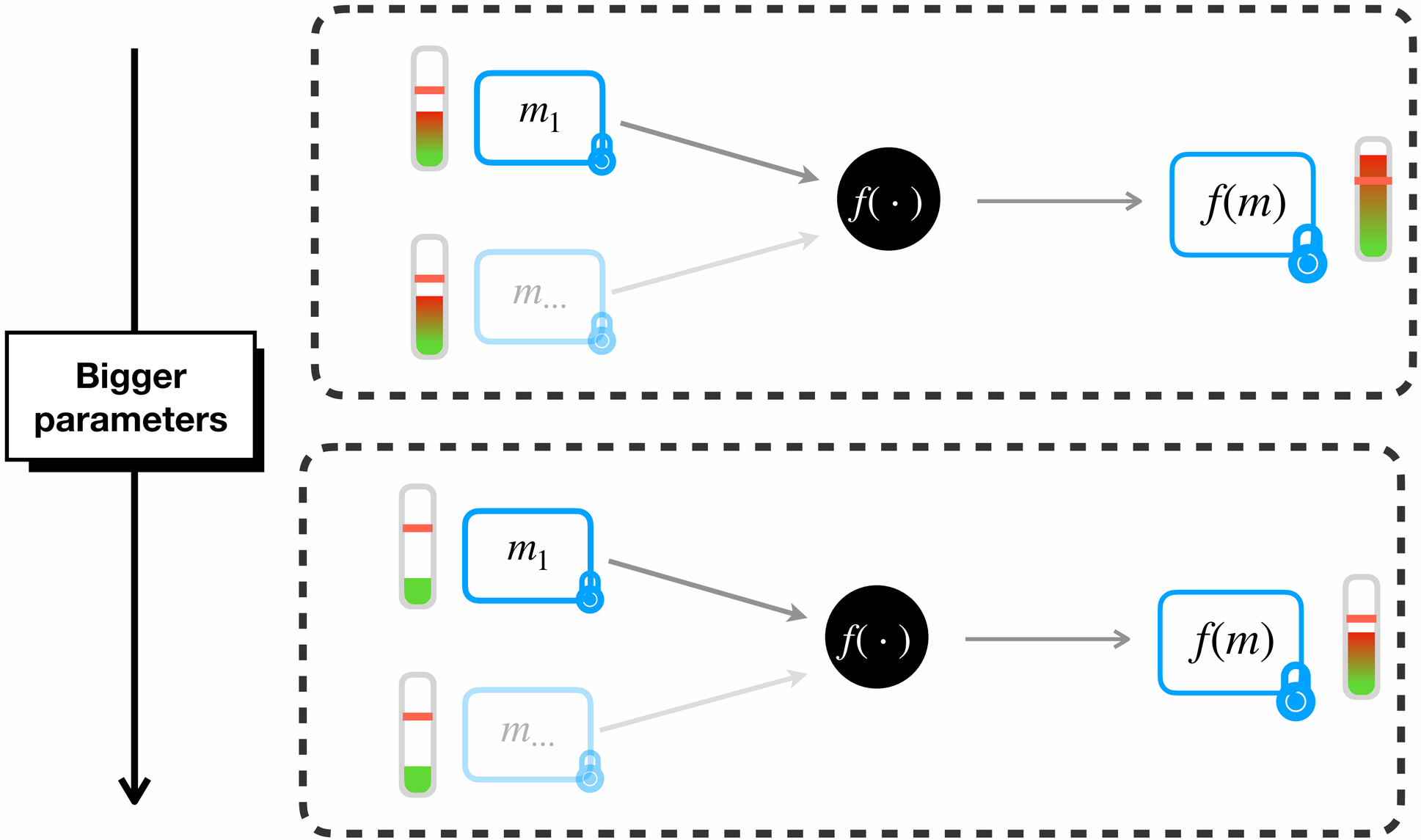
# Leveled Homomorphic Encryption

# Leveled Homomorphic Encryption

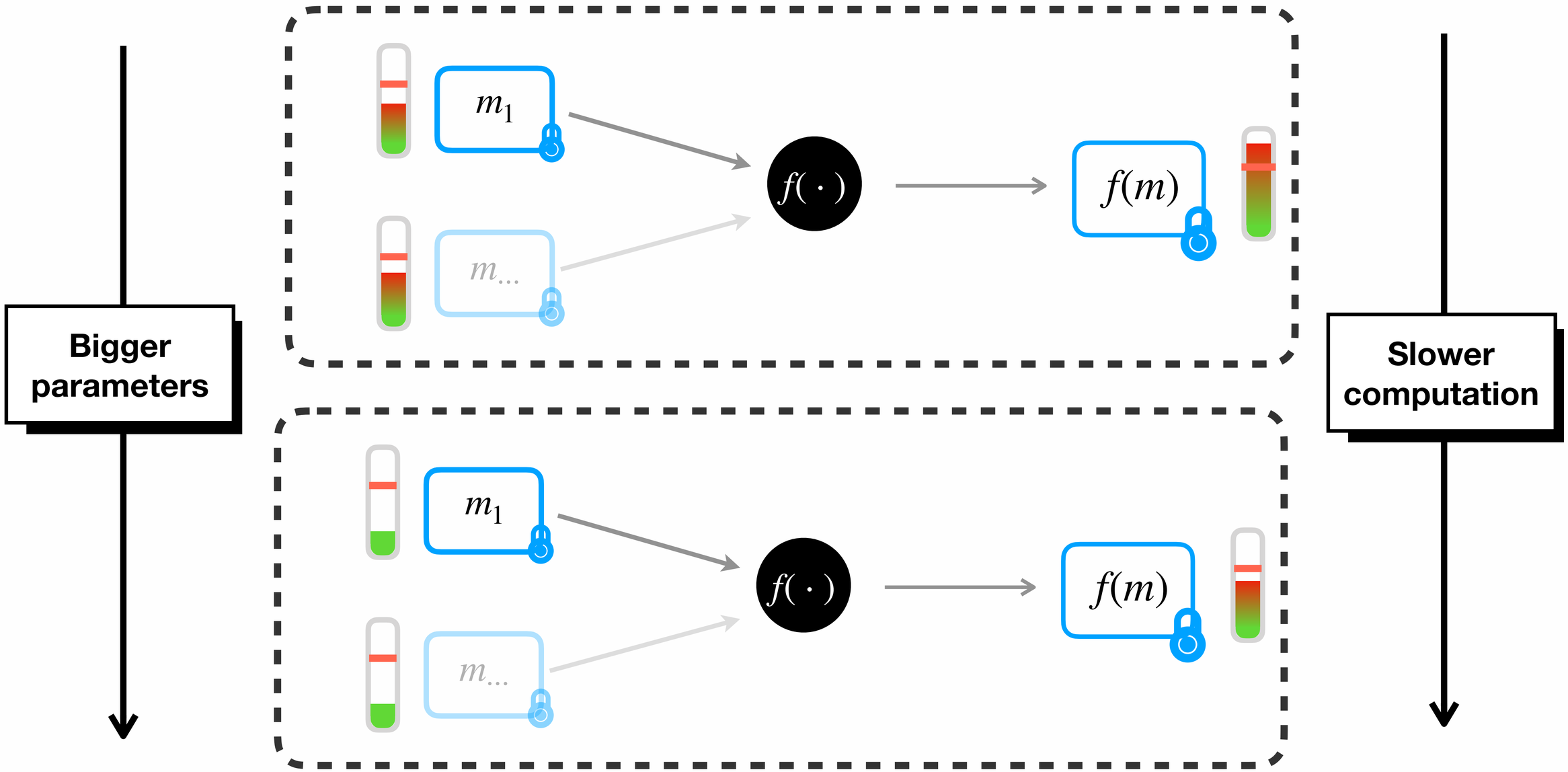
# Leveled Homomorphic Encryption



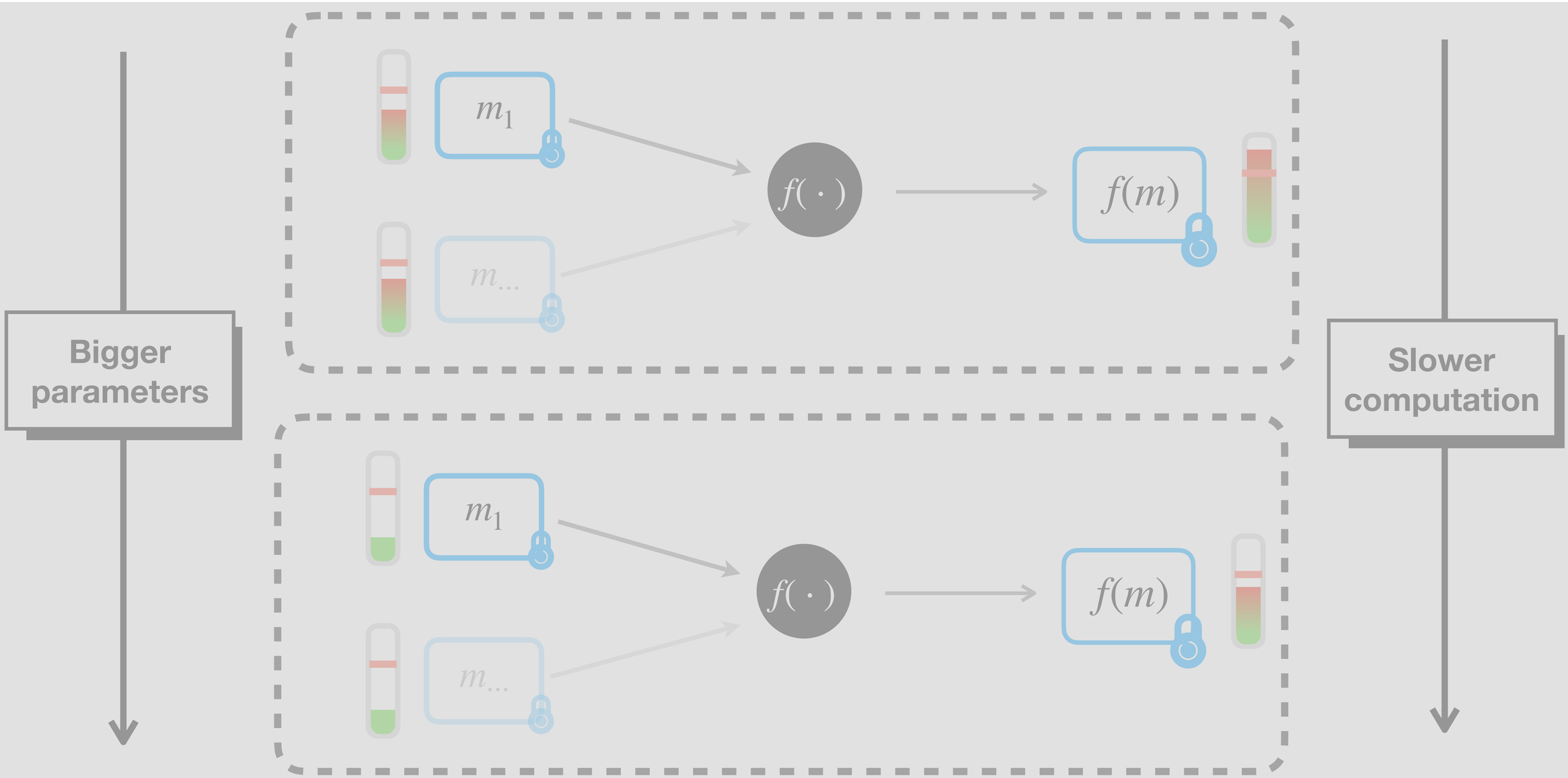
# Leveled Homomorphic Encryption



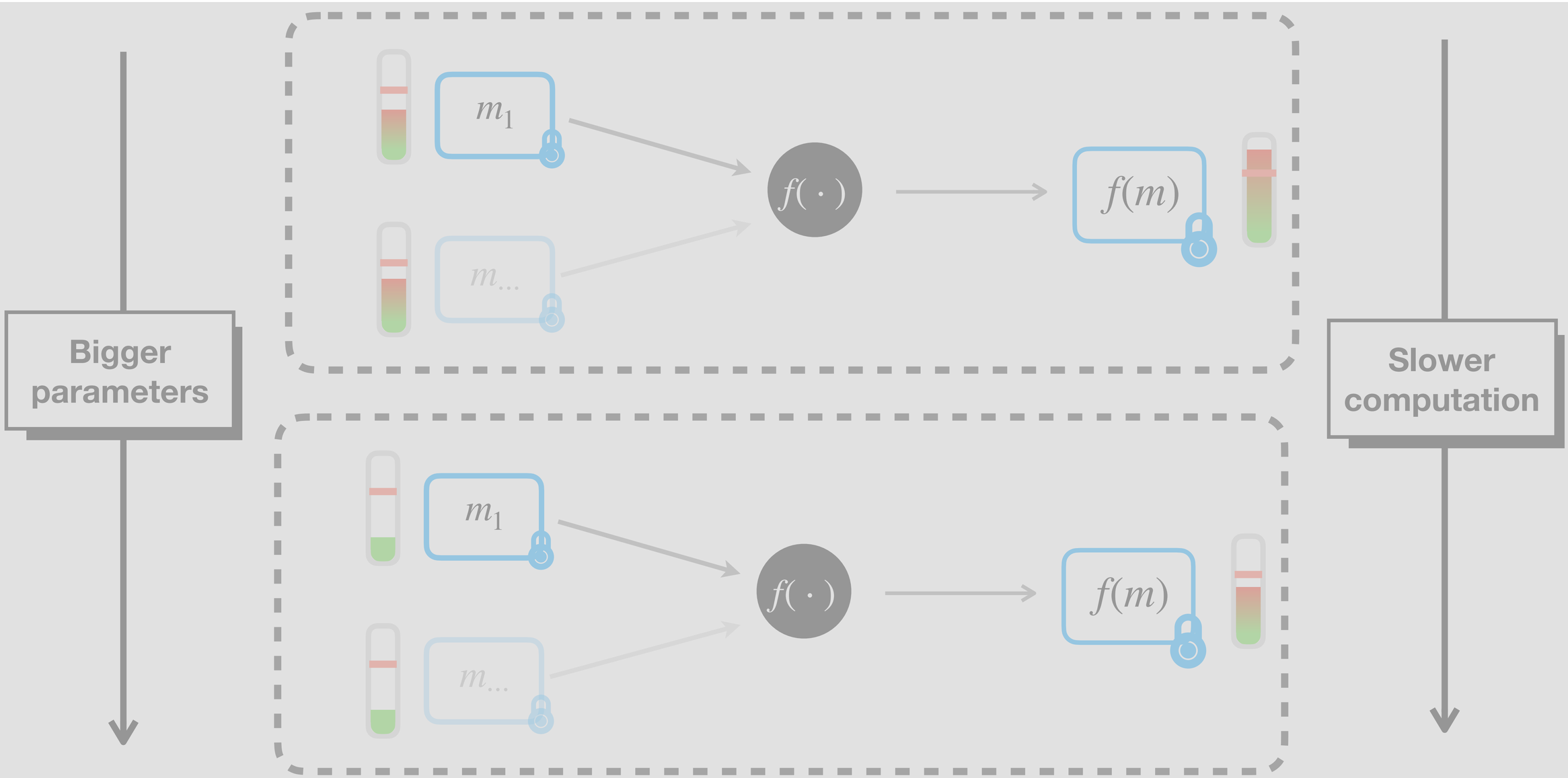
# Leveled Homomorphic Encryption



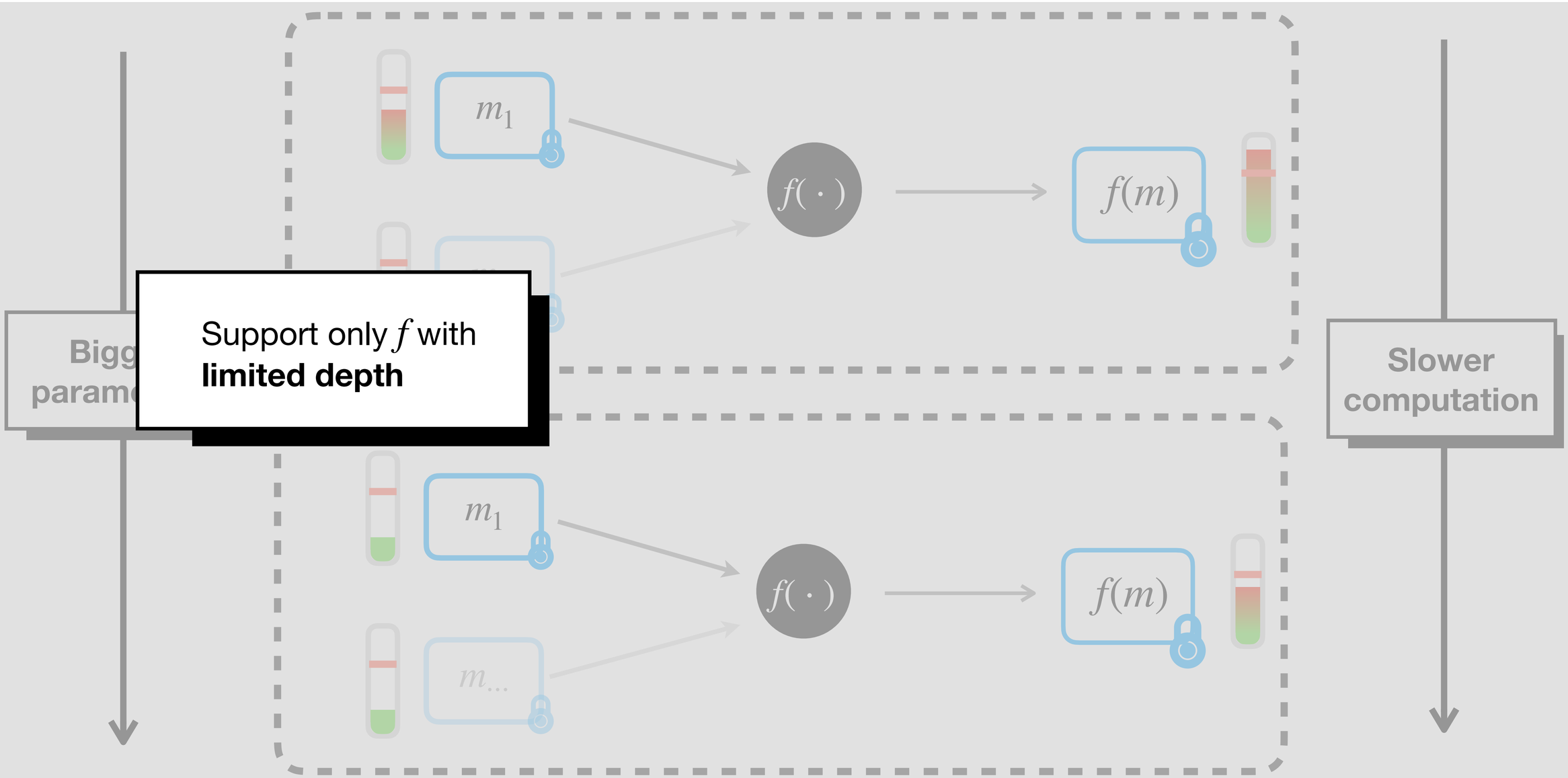
# Leveled Homomorphic Encryption



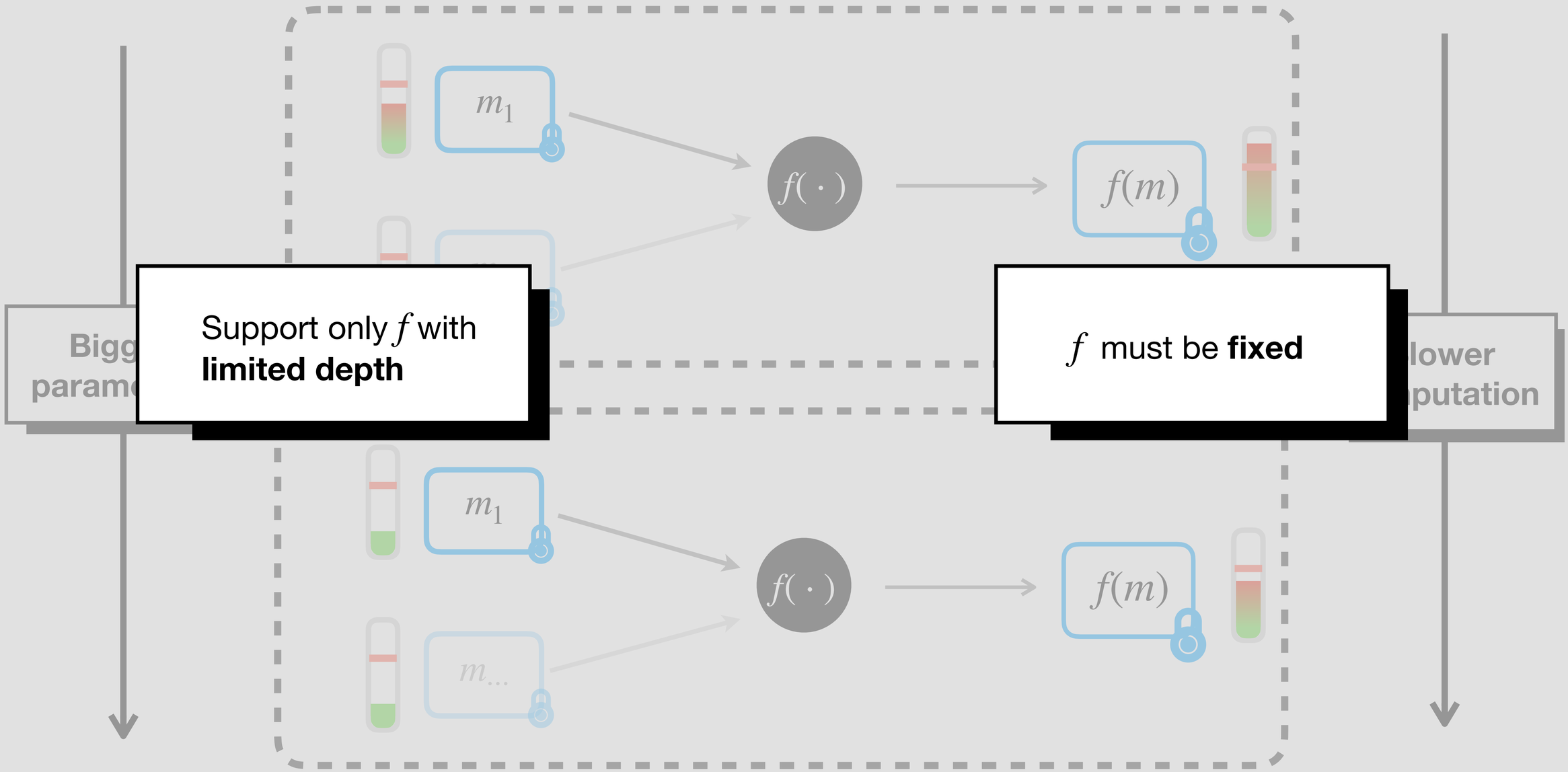
# Leveled Homomorphic Encryption



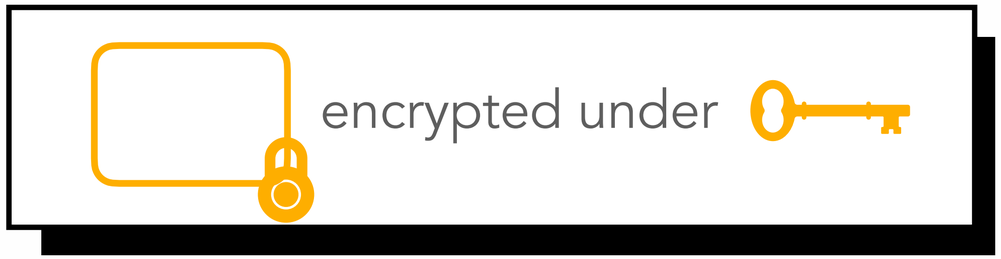
# Leveled Homomorphic Encryption

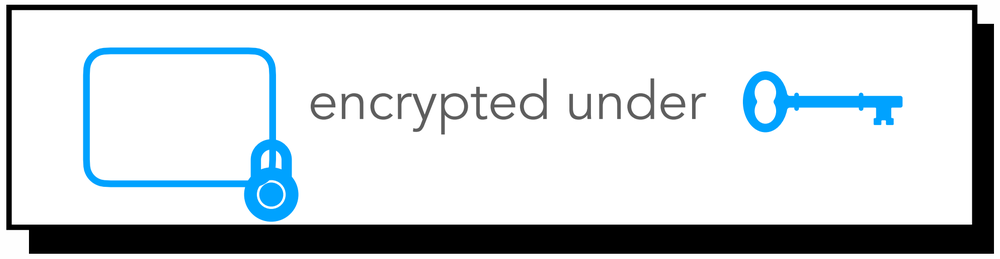


# Leveled Homomorphic Encryption



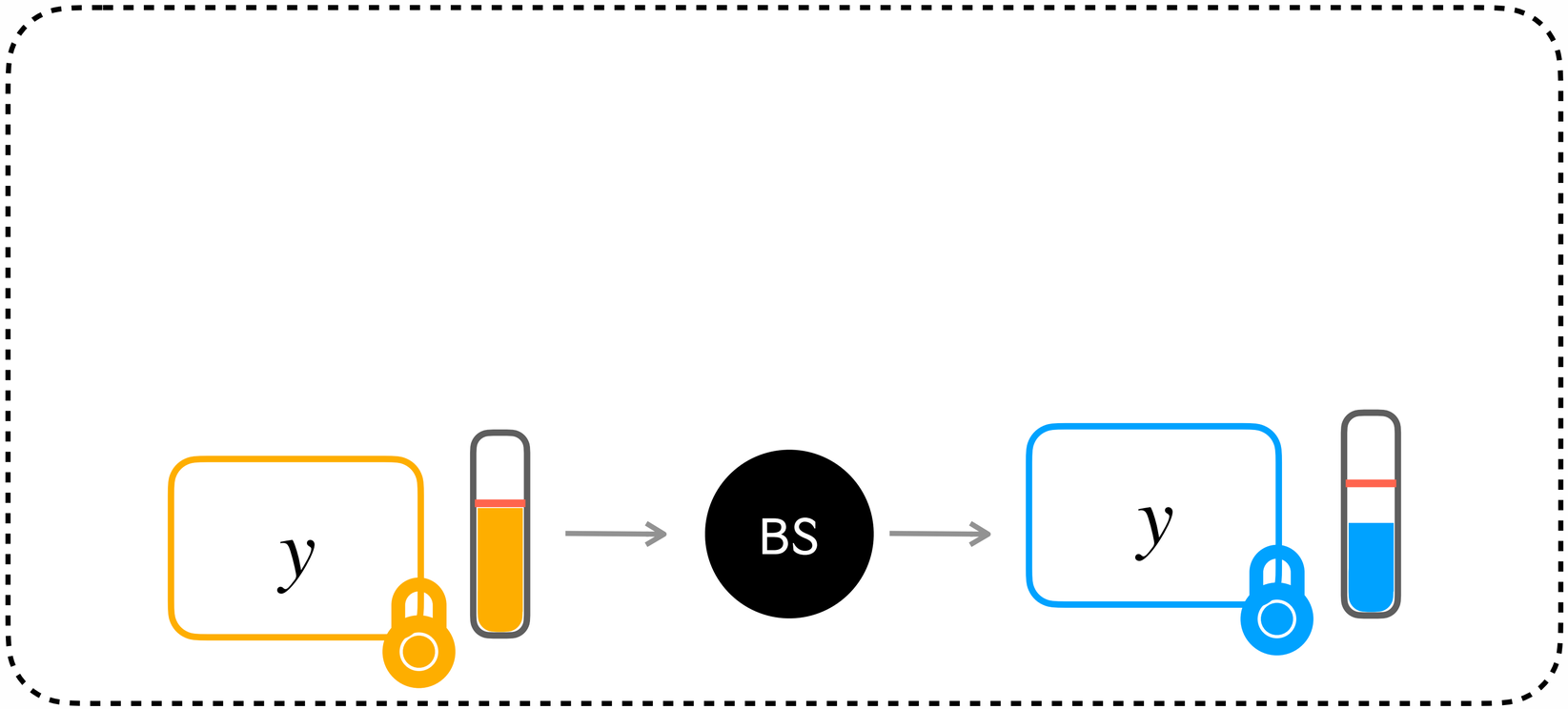
# Bootstrapping [Gen09]

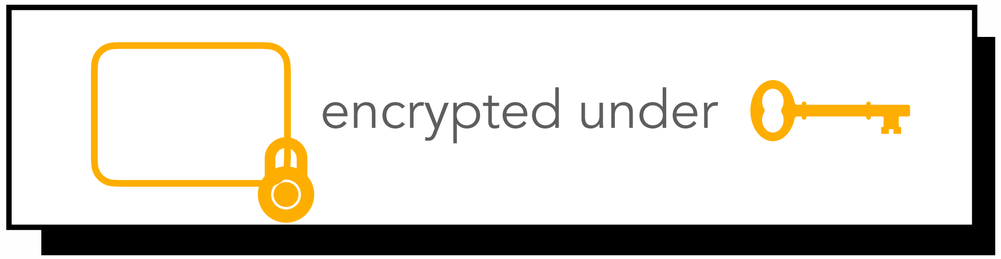


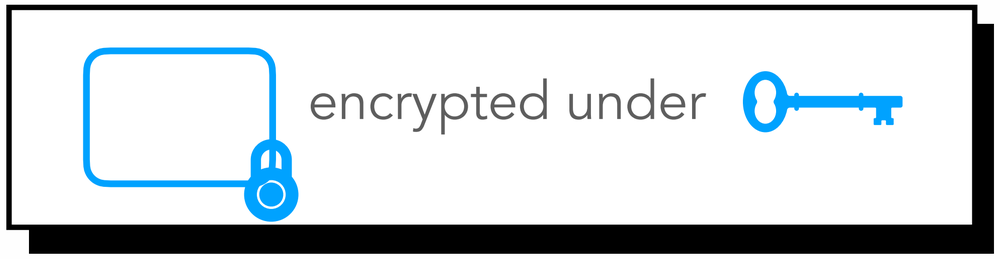




**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# Bootstrapping [Gen09]

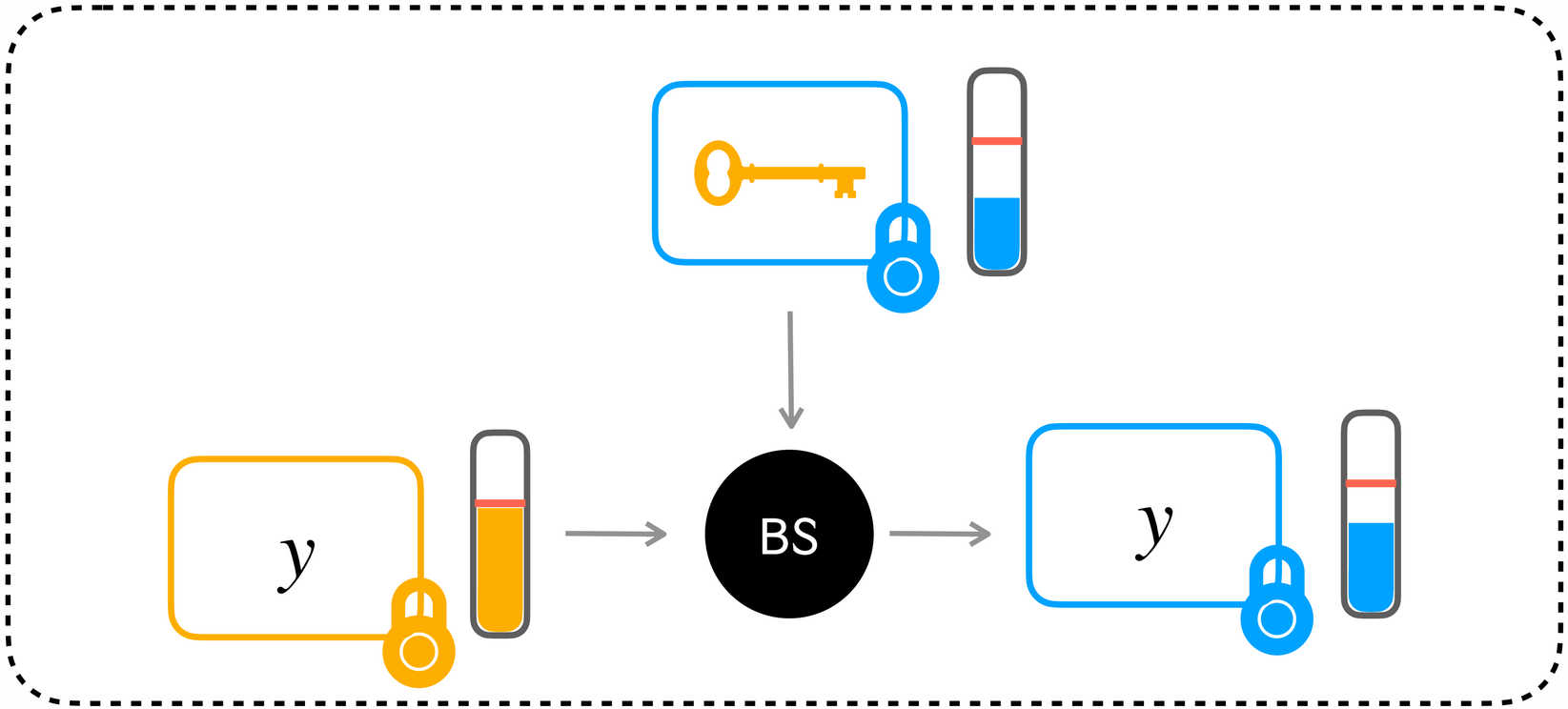


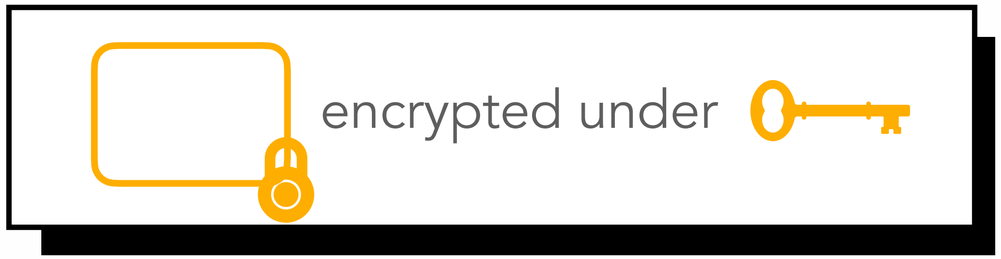



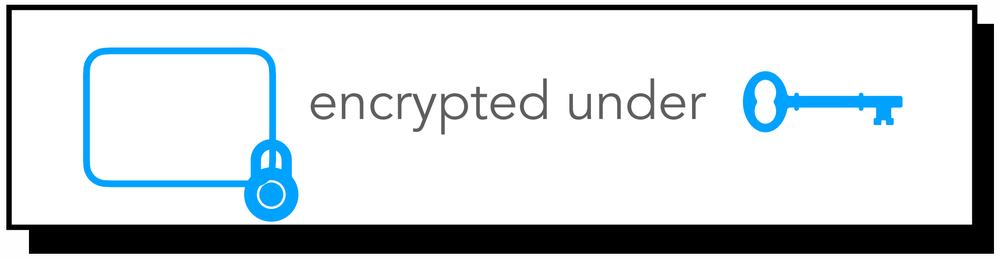




**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# Bootstrapping [Gen09]

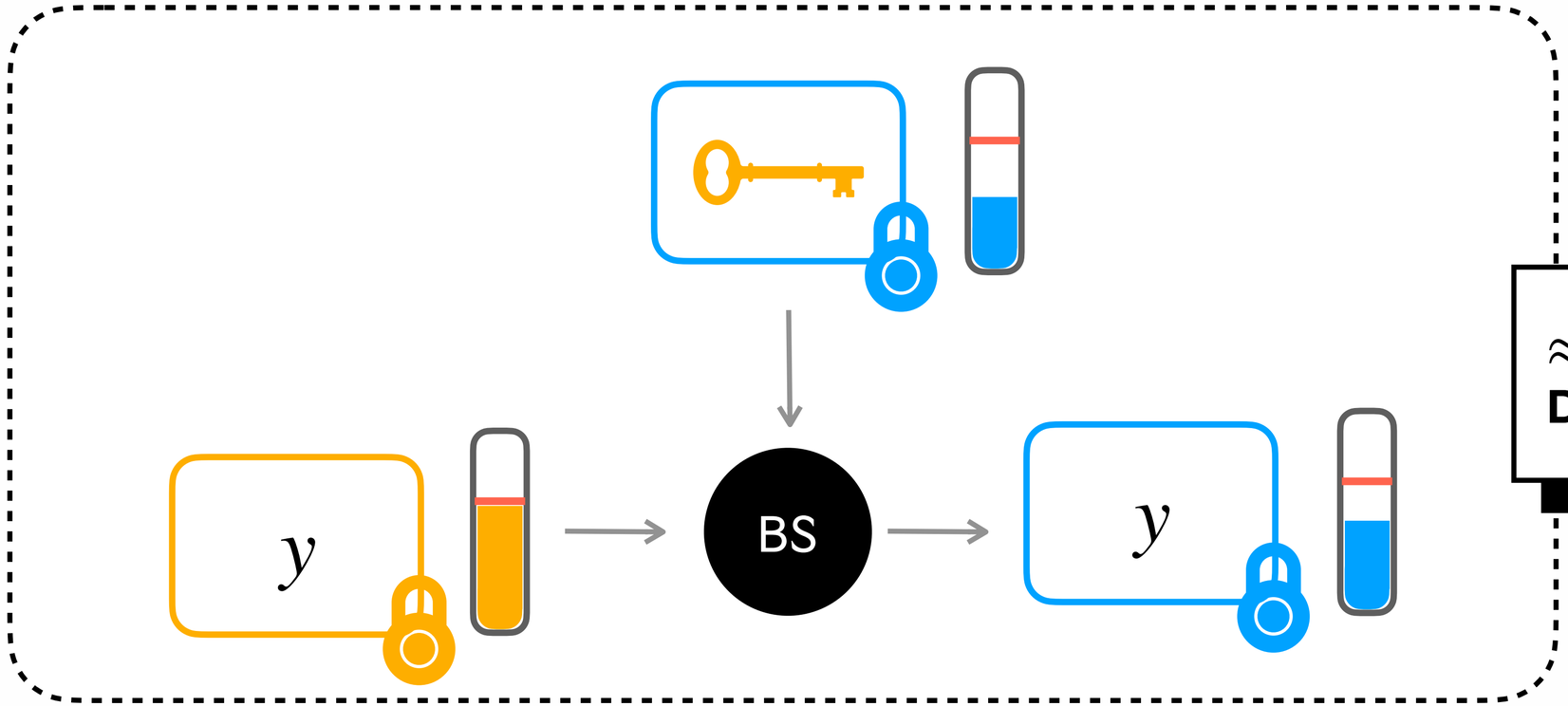


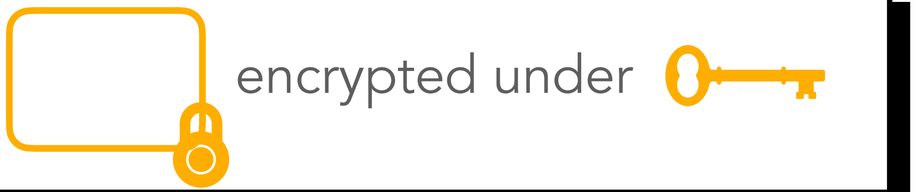



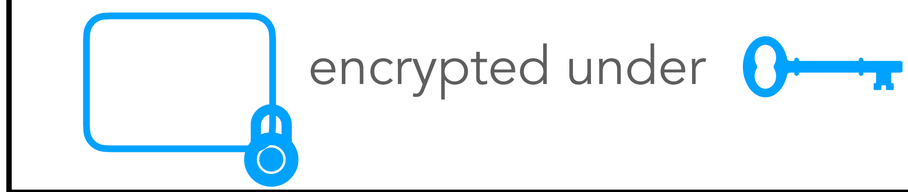

**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# Bootstrapping [Gen09]



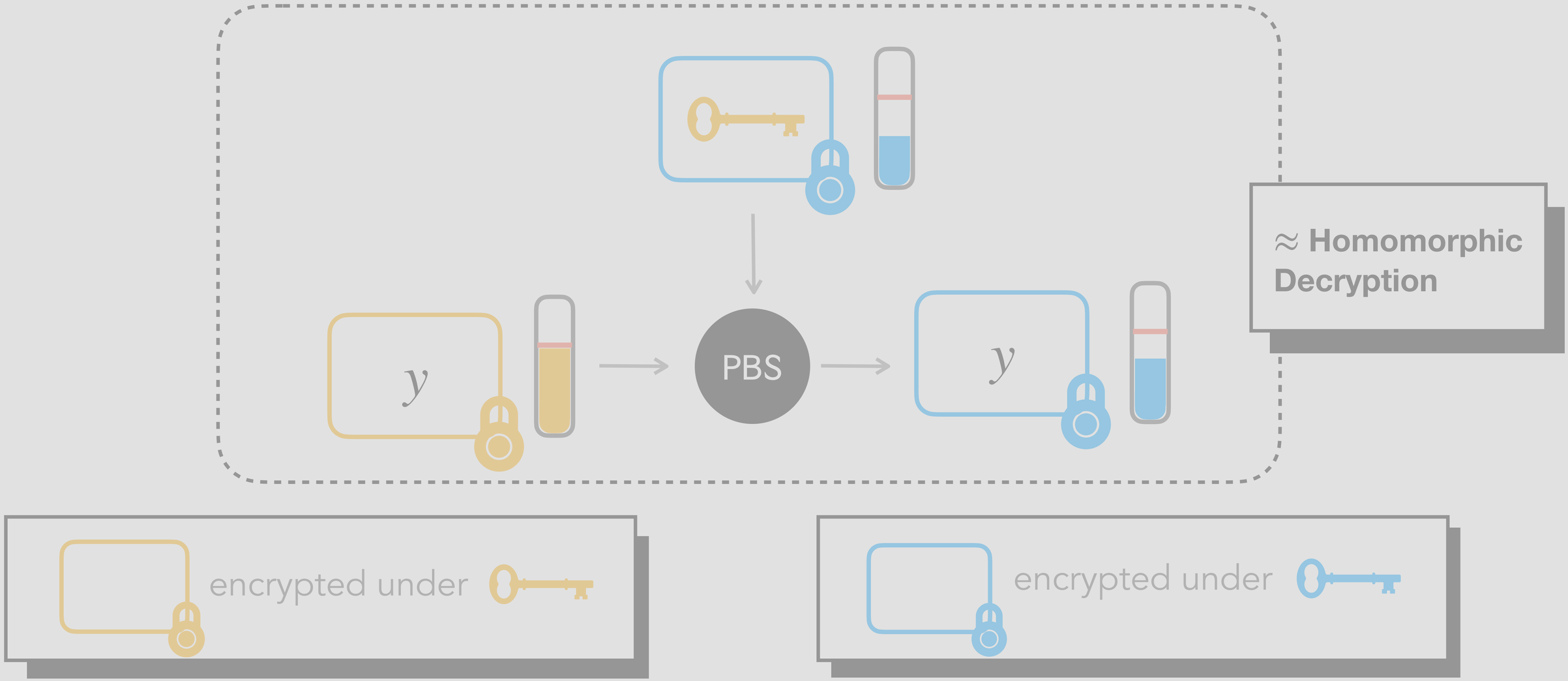

≈ Homomorphic Decryption



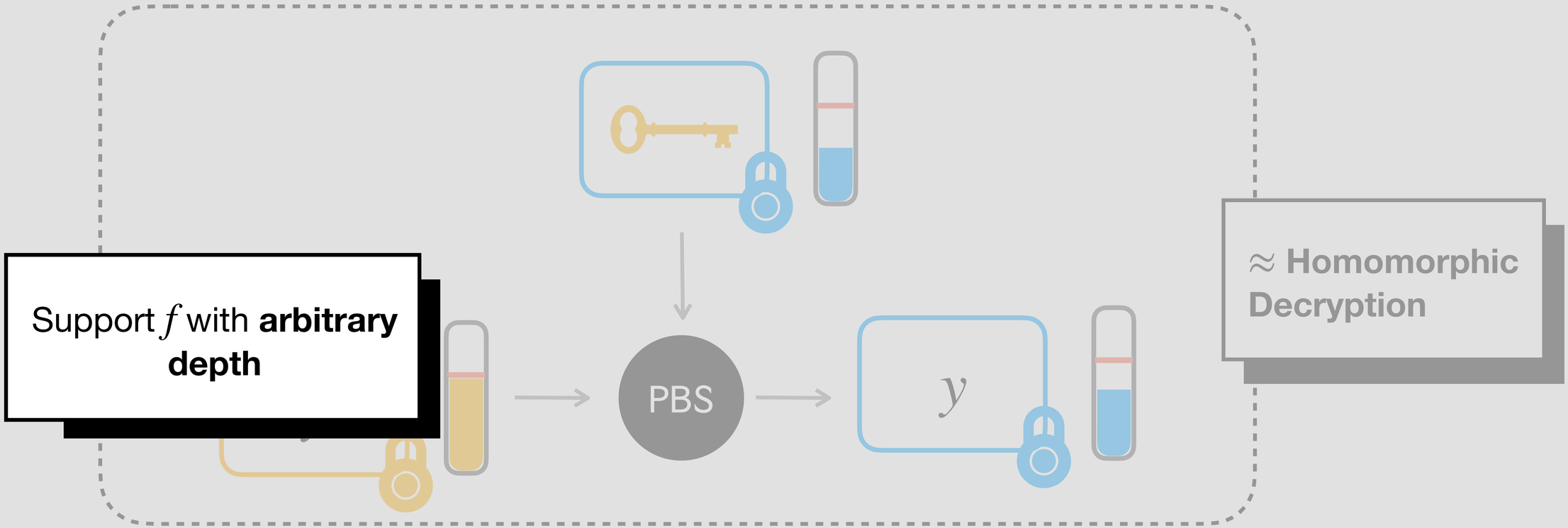

**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

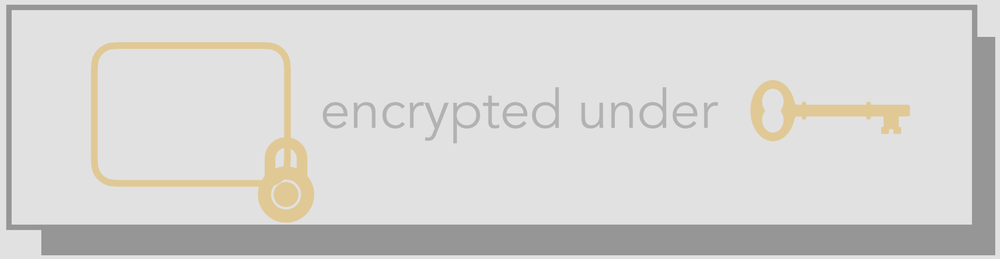
# Bootstrapping [Gen09]

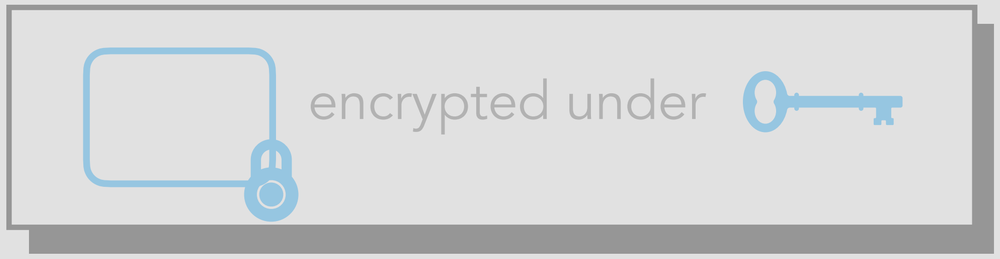



**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# Bootstrapping [Gen09]

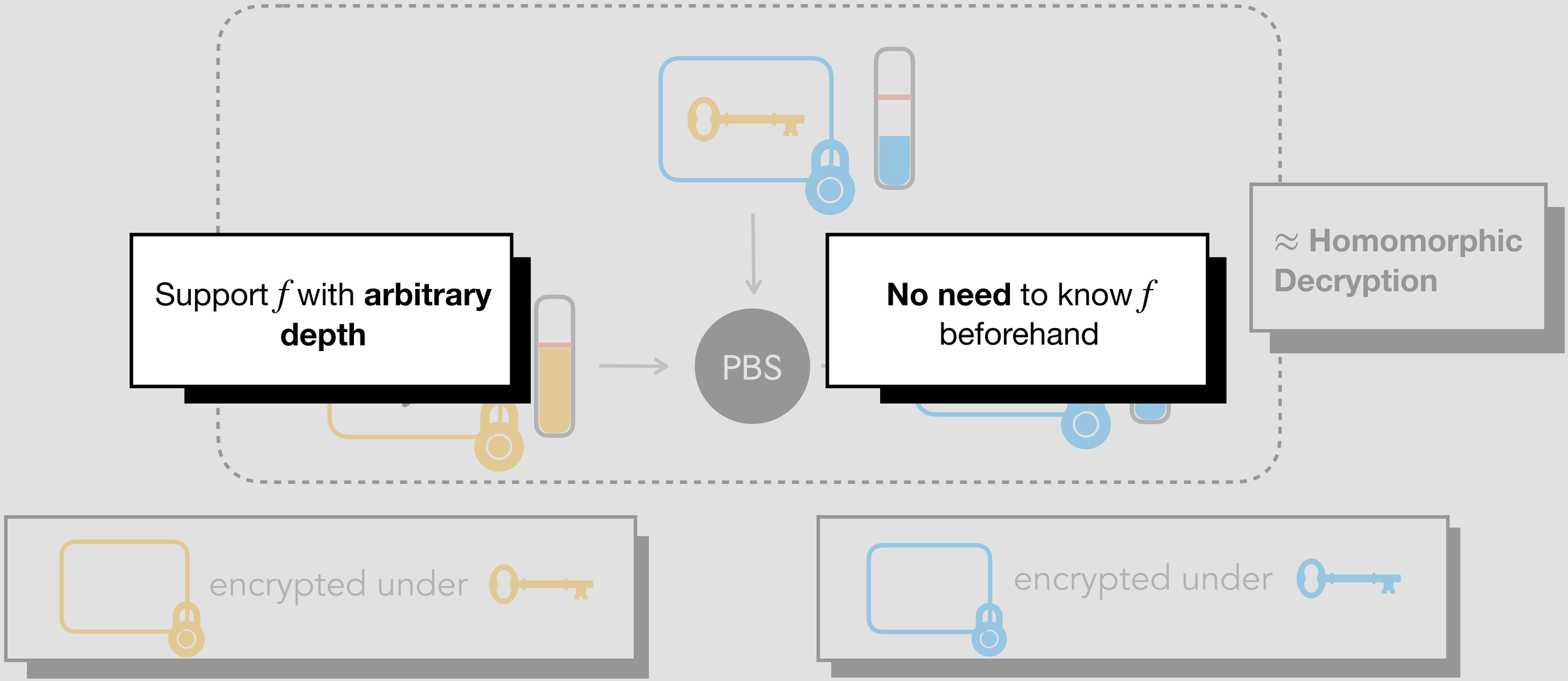



**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# Bootstrapping [Gen09]

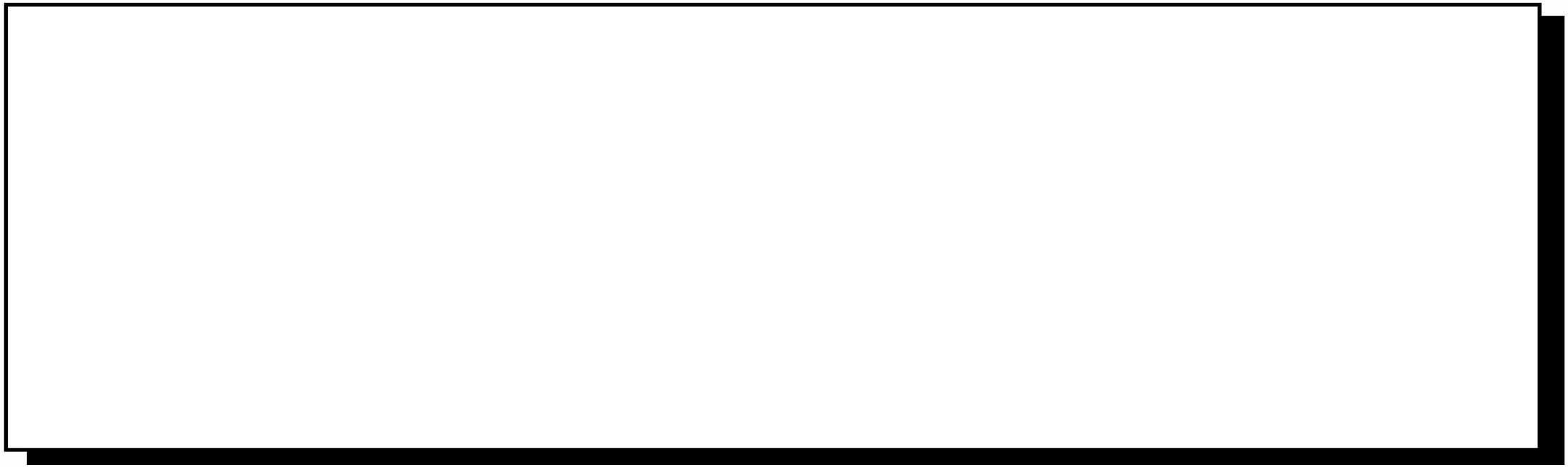



**[Gen09]** C. Gentry. Fully homomorphic encryption using ideal lattices. STOC 2009.

# TFHE

Intuition and main algorithms

# Encryption



Δ

MSB LSB



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathbb{U}([\![0; q-1]\!])$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathbb{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0, \sigma^2)$

$\Delta$

MSB LSB



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathbb{U}([\![0; q-1]\!])$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathbb{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0,\sigma^2)$
2. $b = \langle \vec{a}, \vec{s} \rangle + \Delta m + e \pmod{q}$

$\Delta$

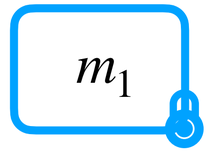
MSB LSB



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathbb{U}(\llbracket 0; q-1 \rrbracket)$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathbb{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0, \sigma^2)$
2. $b = \langle \vec{a}, \vec{s} \rangle + \Delta m + e \pmod{q}$




$= (\vec{a}, b)$

Δ

MSB LSB



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

LWE Dimension

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathbb{U}(\llbracket 0; q-1 \rrbracket)$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathbb{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0, \sigma^2)$
2. $b = \langle \vec{a}, \vec{s} \rangle + \Delta m + e \pmod{q}$

$m_1$ $= (\vec{a}, b)$

$\Delta$

MSB LSB



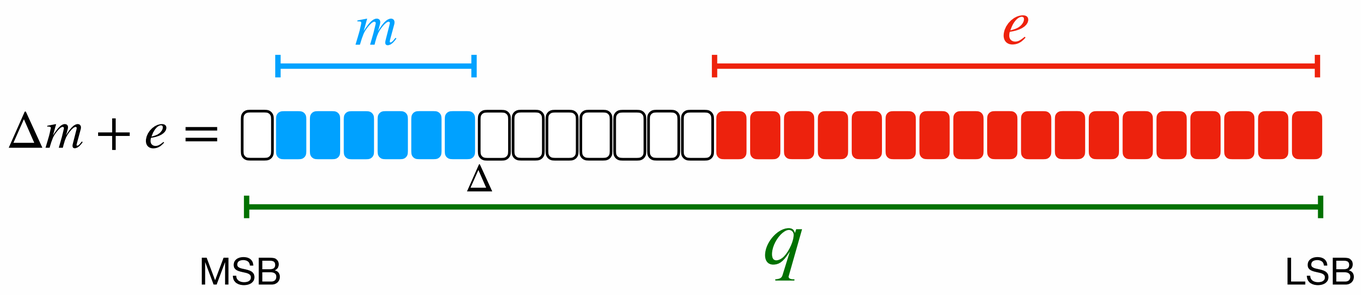
**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

LWE Dimension

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathcal{U}(\llbracket 0; q-1 \rrbracket)$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathcal{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0, \sigma^2)$
2. $b = \langle \vec{a}, \vec{s} \rangle + \Delta m + e \pmod{q}$

$m_1$ $= (\vec{a}, b)$





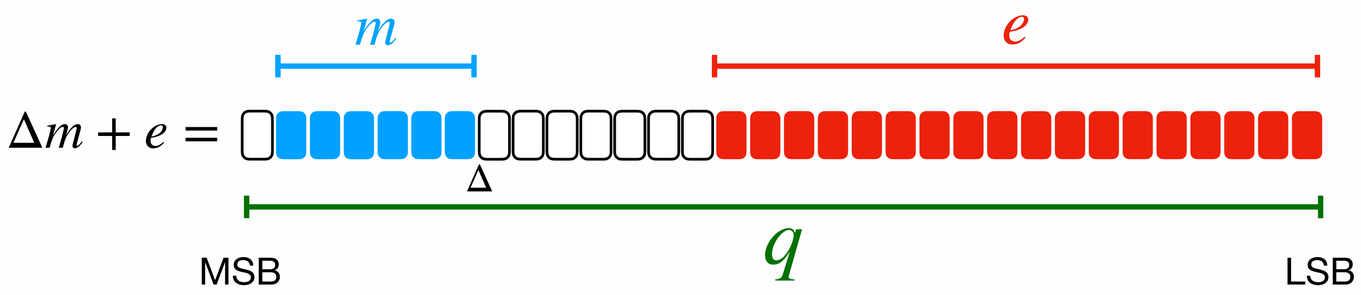

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Encryption

LWE Dimension

1. Draw a mask $\vec{a} = (a_0, \cdots, a_{n-1})$ from $\mathbb{U}(\llbracket 0; q-1 \rrbracket)$ and
   a secret key $\vec{s} = (s_0, \cdots, s_{n-1})$ from $\mathbb{U}(\{0,1\})$
   a noise $e$ from $\mathcal{N}(0, \sigma^2)$
2. $b = \langle \vec{a}, \vec{s} \rangle + \Delta m + e \pmod q$

$m_1$ $= (\vec{a}, b)$




For a security level $\lambda$,

$n \nearrow \implies \sigma \searrow$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Decryption



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Decryption

1 $$b - \langle \vec{a}, \vec{s} \rangle = \cancel{\langle \vec{a}, \vec{s} \rangle} + \Delta m + e - \cancel{\langle \vec{a}, \vec{s} \rangle} \pmod{q}$$

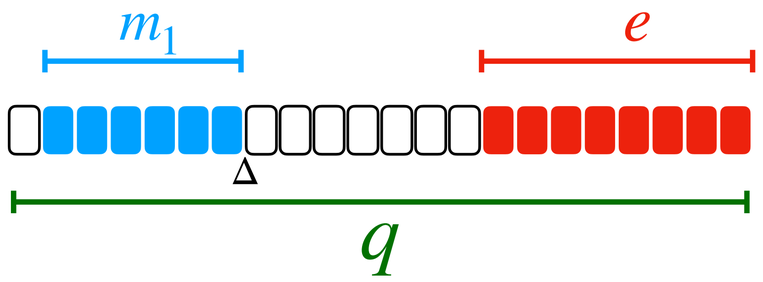


**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Decryption

1 $$b - \langle \vec{a}, \vec{s} \rangle = \cancel{\langle \vec{a}, \vec{s} \rangle} + \Delta m + e - \cancel{\langle \vec{a}, \vec{s} \rangle} \pmod{q}$$

2a $$|e| < \frac{\Delta}{2} \Longrightarrow \left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$$



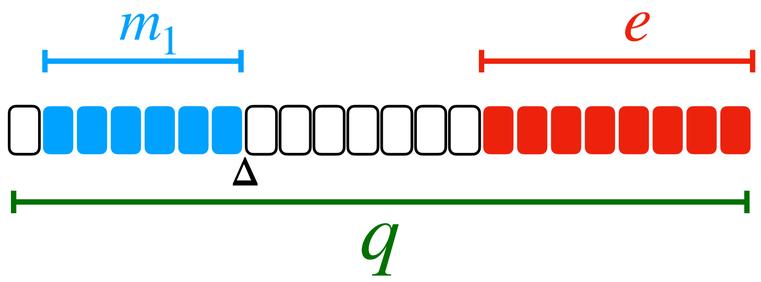



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

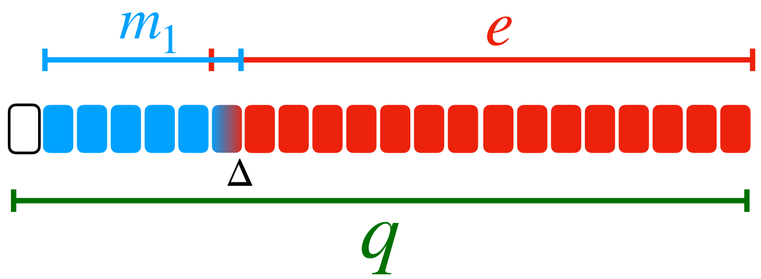
# Decryption

1 $$b - \langle \vec{a}, \vec{s} \rangle = \cancel{\langle \vec{a}, \vec{s} \rangle} + \Delta m + e - \cancel{\langle \vec{a}, \vec{s} \rangle} \pmod{q}$$

2a $$|e| < \frac{\Delta}{2} \Longrightarrow \left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$$




2b $$|e| \geq \frac{\Delta}{2} \Longrightarrow \left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor \neq m$$





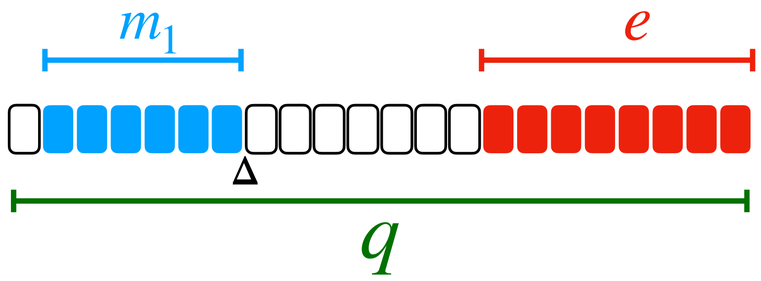

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Decryption

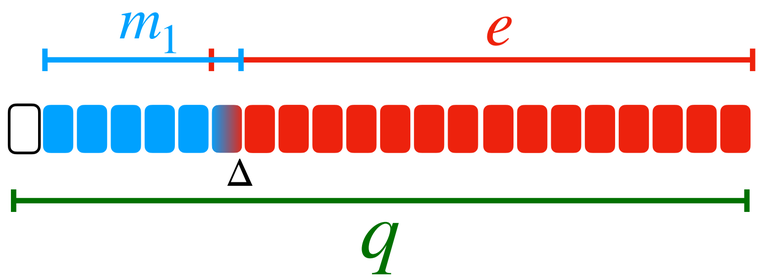
**Condition to guarantee the correctness of a computation**

1 $$b - \langle \vec{a}, \vec{s} \rangle = \cancel{\langle \vec{a}, \vec{s} \rangle} + \Delta m + e - \cancel{\langle \vec{a}, \vec{s} \rangle} \pmod{q}$$

2a $$|e| < \frac{\Delta}{2} \Longrightarrow \left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$$




2b $$|e| \geq \frac{\Delta}{2} \Longrightarrow \left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor \neq m$$






**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\overrightarrow{a_1}, b_1)$ and $\mathsf{ct}_2 = (\overrightarrow{a_2}, b_2)$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\overrightarrow{a_1}, b_1)$ and $\mathsf{ct}_2 = (\overrightarrow{a_2}, b_2)$

Define $\mathsf{ct}_3$ such that

$$\mathsf{ct}_3 \leftarrow \mathsf{ct}_1 \oplus \mathsf{ct}_2 = \left(\overrightarrow{a_1} + \overrightarrow{a_2} \pmod q, b_1 + b_2 \pmod q\right)$$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\overrightarrow{a_1}, b_1)$ and $\mathsf{ct}_2 = (\overrightarrow{a_2}, b_2)$

Define $\mathsf{ct}_3$ such that

$$\mathsf{ct}_3 \leftarrow \mathsf{ct}_1 \oplus \mathsf{ct}_2 = \left(\overrightarrow{a_1} + \overrightarrow{a_2} \pmod q, b_1 + b_2 \pmod q\right)$$

Decrypt $\mathsf{ct}_3$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\overrightarrow{a_1}, b_1)$ and $\mathsf{ct}_2 = (\overrightarrow{a_2}, b_2)$

Define $\mathsf{ct}_3$ such that

$$\mathsf{ct}_3 \leftarrow \mathsf{ct}_1 \oplus \mathsf{ct}_2 = \left(\overrightarrow{a_1} + \overrightarrow{a_2} \pmod q, b_1 + b_2 \pmod q\right)$$

Decrypt $\mathsf{ct}_3$

$$b_1 + b_2 - < \overrightarrow{a_1} + \overrightarrow{a_2}, \vec{s} > \pmod q$$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\vec{a_1}, b_1)$ and $\mathsf{ct}_2 = (\vec{a_2}, b_2)$

Define $\mathsf{ct}_3$ such that

$$\mathsf{ct}_3 \leftarrow \mathsf{ct}_1 \oplus \mathsf{ct}_2 = \left(\vec{a_1} + \vec{a_2} \pmod q, b_1 + b_2 \pmod q\right)$$

Decrypt $\mathsf{ct}_3$

$$b_1 + b_2 - < \vec{a_1} + \vec{a_2}, \vec{s} > \pmod q$$

$$= \cancel{\left\langle \vec{a_1} + \vec{a_2}, \vec{s} \right\rangle} + \underbrace{m_1 + m_2}_{\text{message}} + \underbrace{e_1 + e_2}_{\text{noise}} - \cancel{\left\langle \vec{a_1} + \vec{a_2}, \vec{s} \right\rangle} \pmod q$$



**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Addition

Consider $\mathsf{ct}_1 = (\vec{a_1}, b_1)$ and $\mathsf{ct}_2 = (\vec{a_2}, b_2)$

Define $\mathsf{ct}_3$ such that

$$\mathsf{ct}_3 \leftarrow \mathsf{ct}_1 \oplus \mathsf{ct}_2 = \left(\vec{a_1} + \vec{a_2} \pmod q, b_1 + b_2 \pmod q\right)$$

Decrypt $\mathsf{ct}_3$

$$b_1 + b_2 - < \vec{a_1} + \vec{a_2}, \vec{s} > \pmod q$$

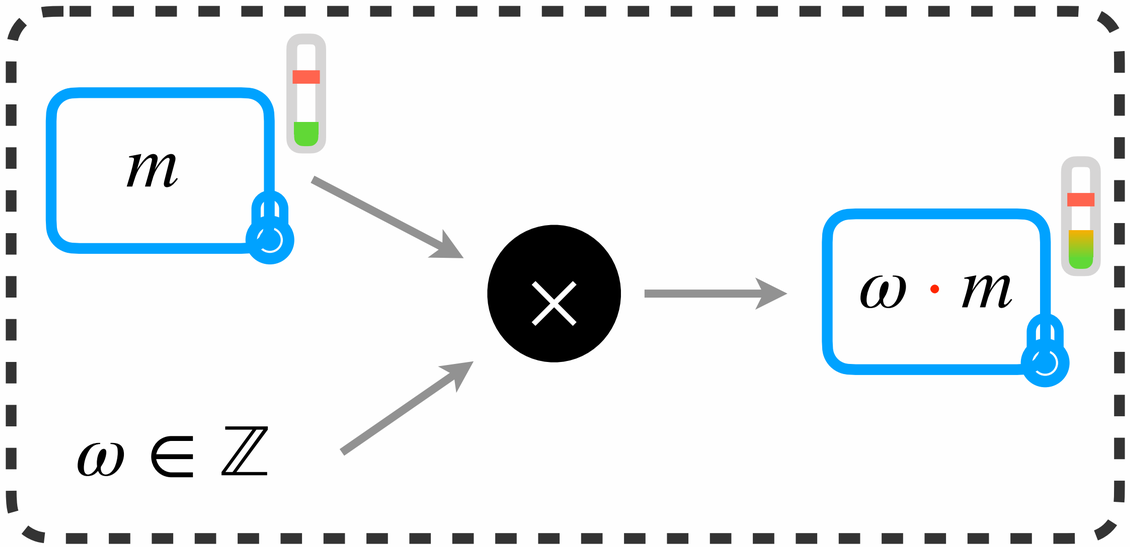
$$= \cancel{\left\langle \vec{a_1} + \vec{a_2}, \vec{s} \right\rangle} + \underbrace{m_1 + m_2}_{\text{message}} + \underbrace{e_1 + e_2}_{\text{noise}} - \cancel{\left\langle \vec{a_1} + \vec{a_2}, \vec{s} \right\rangle} \pmod q$$



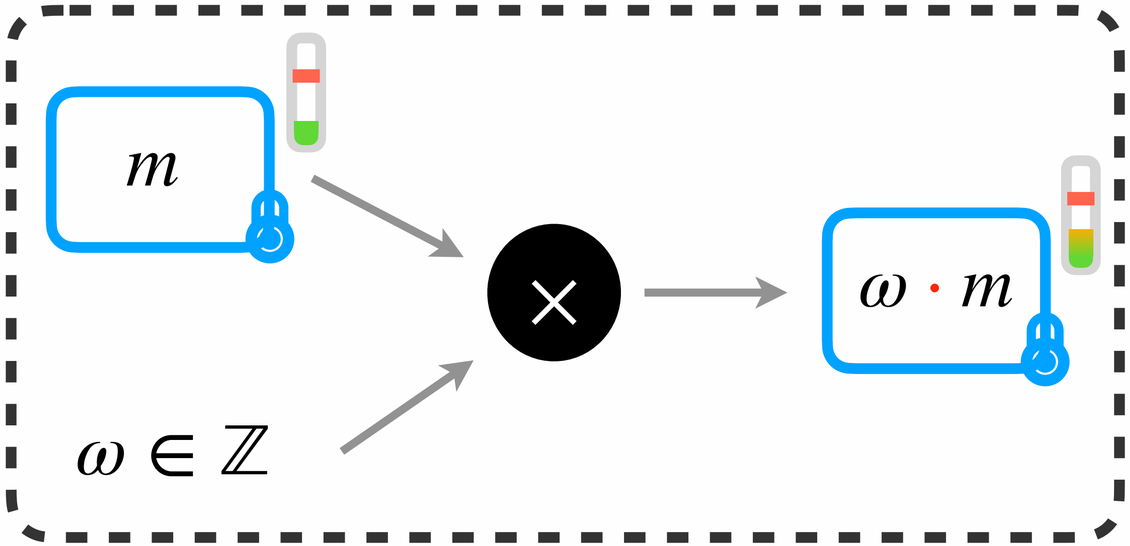

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

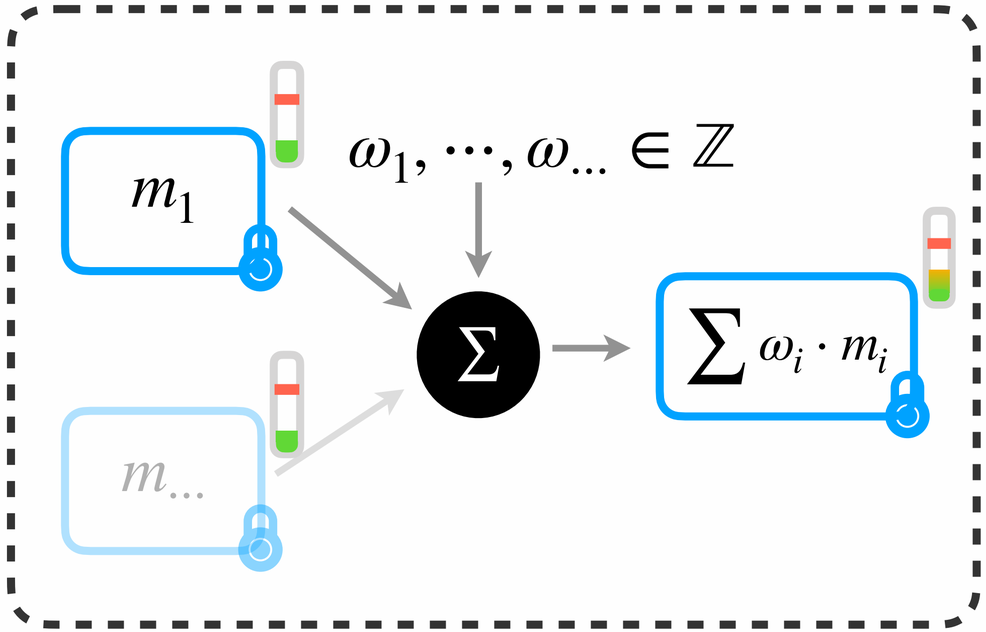
# Dot Product

# Dot Product



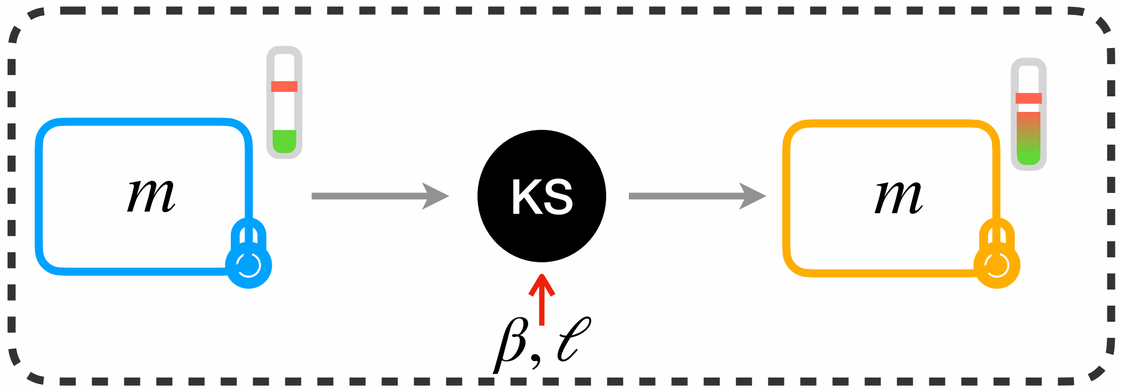
# Dot Product


$m$
$\omega \in \mathbb{Z}$
$\times$
$\omega \cdot m$


$m_1$
$m_{...}$
$\omega_1, \cdots, \omega_{...} \in \mathbb{Z}$
$\Sigma$
$\sum \omega_i \cdot m_i$

# Keyswitch

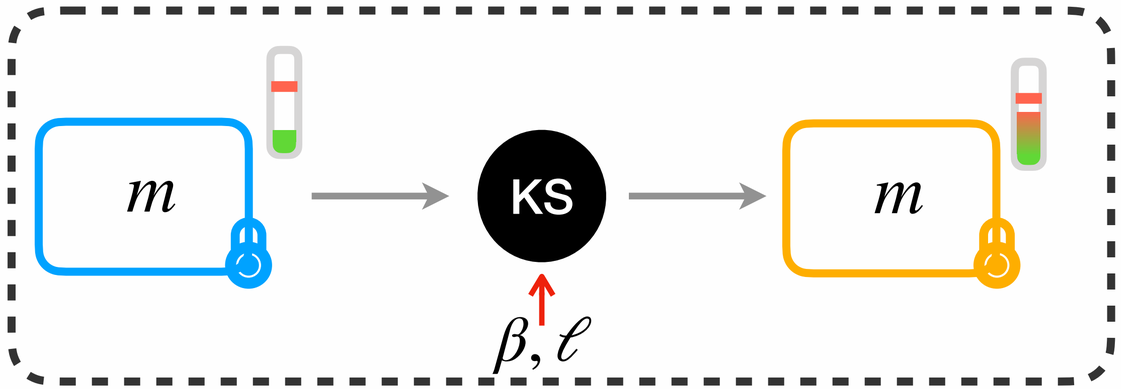
# Keyswitch



# Keyswitch

1. $b - \langle \vec{a}, \vec{s} \rangle = \Delta m + e \pmod{q}$

2. $\left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$

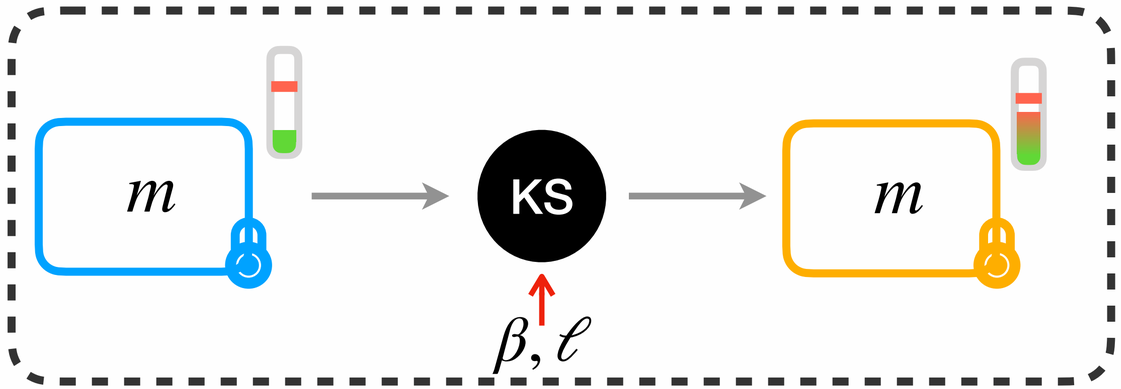


# Keyswitch

1. $b - \langle \vec{a}, \vec{s} \rangle = \Delta m + e \pmod{q}$

2. $\left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$

← Only require homomorphic dot product



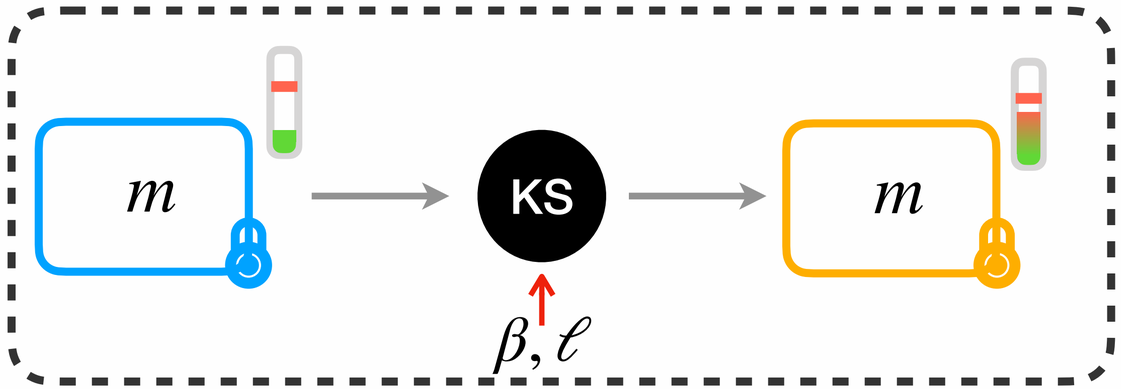
# Keyswitch

Non-linear operation →

1 $b - \langle \vec{a}, \vec{s} \rangle = \Delta m + e \pmod{q}$

2 $\left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$

← Only require homomorphic dot product



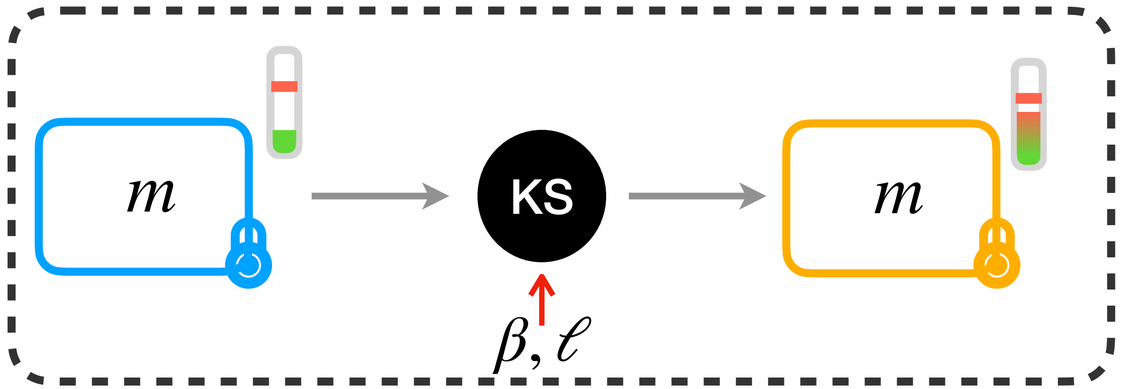
# Keyswitch

1 $b - \langle \vec{a}, \vec{s} \rangle = \Delta m + e \pmod{q}$

2 $\left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$

Non-linear operation →

← Only require homomorphic dot product

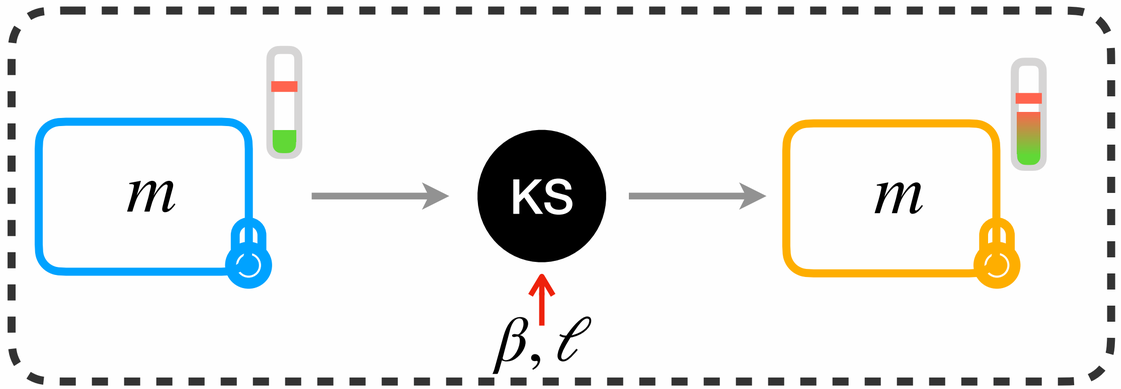


# Keyswitch

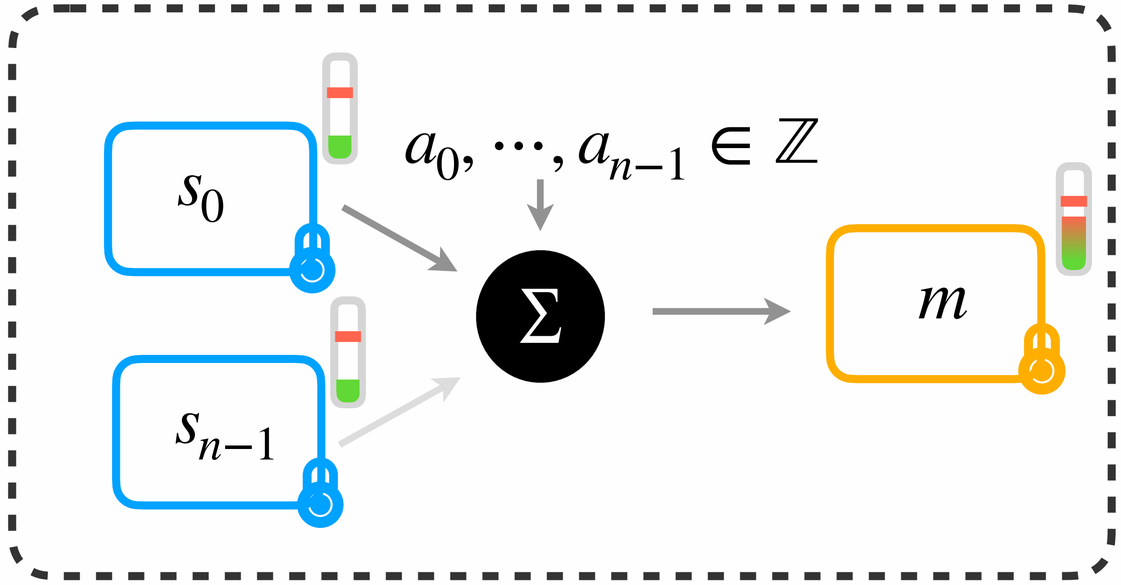
Non-linear operation →

1. $b - \langle \vec{a}, \vec{s} \rangle = \Delta m + e \pmod{q}$
2. $\left\lceil \frac{\Delta m + e}{\Delta} \right\rfloor = m$

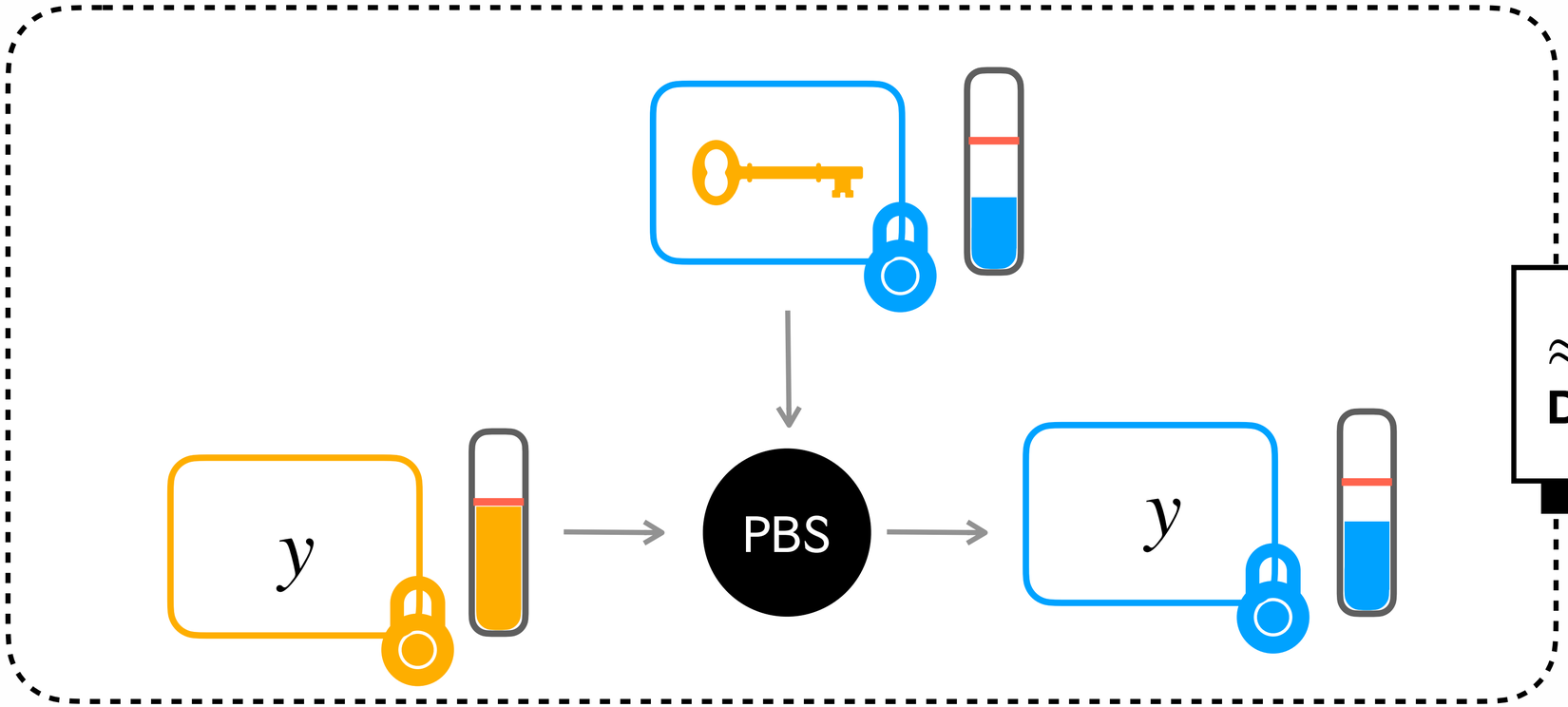
← Only require homomorphic dot product



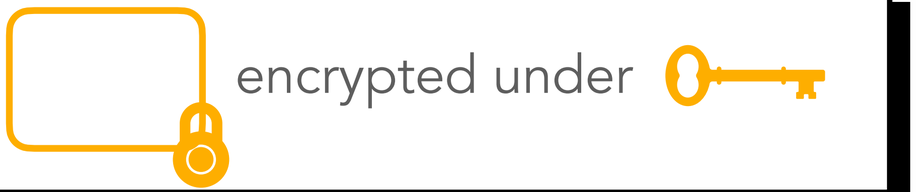

=

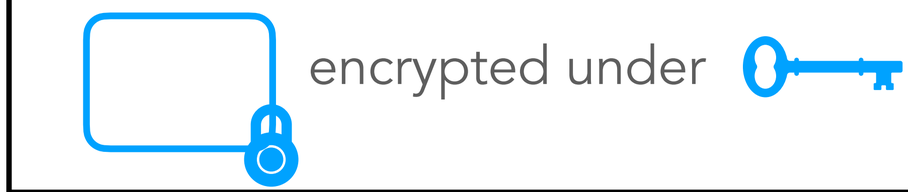


# Programmable Bootstrapping (PBS)




≈ **Homomorphic Decryption**

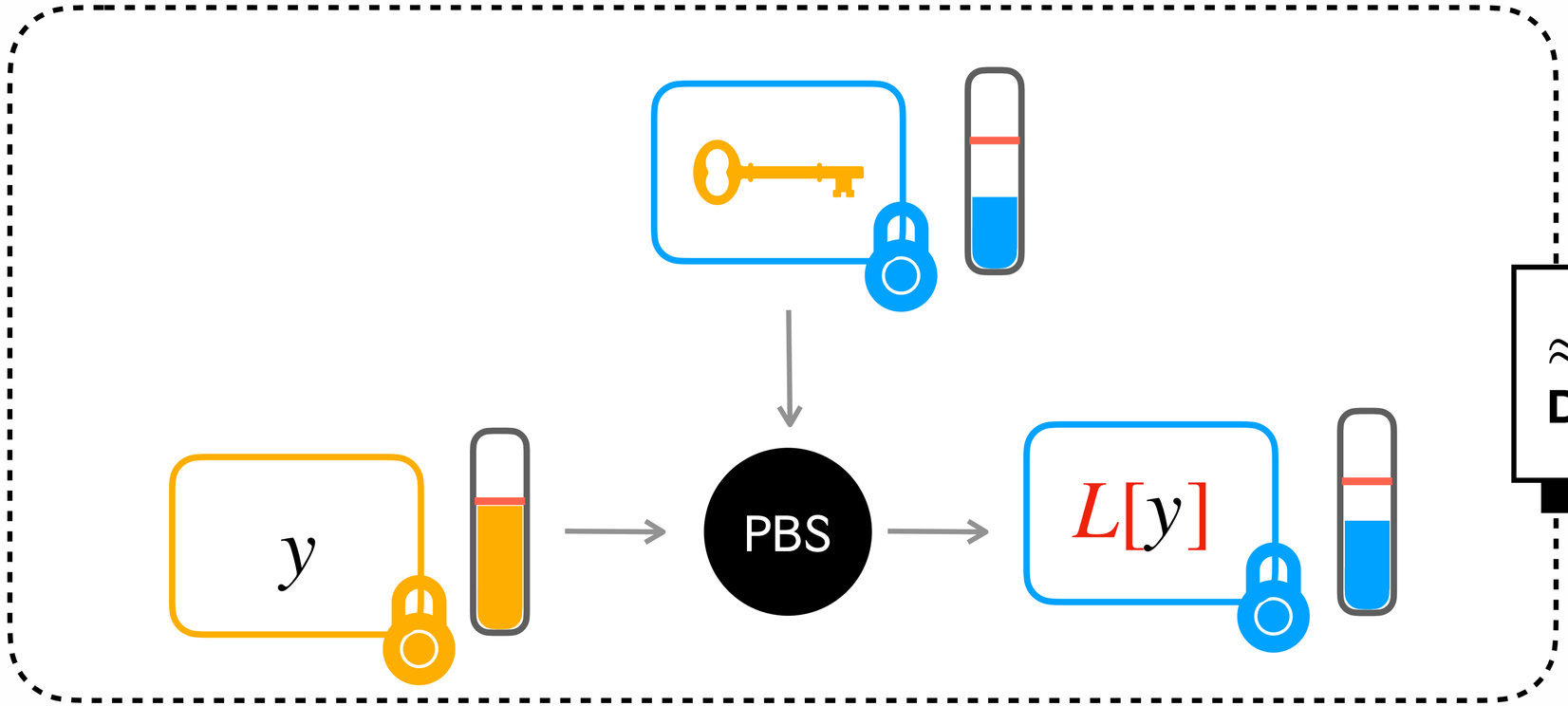


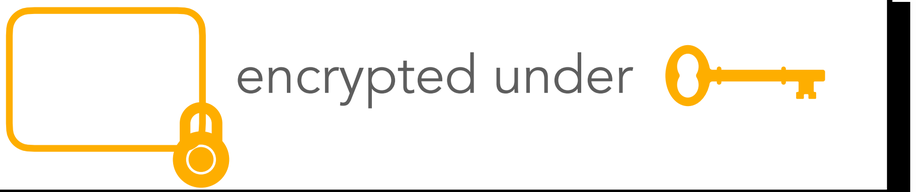


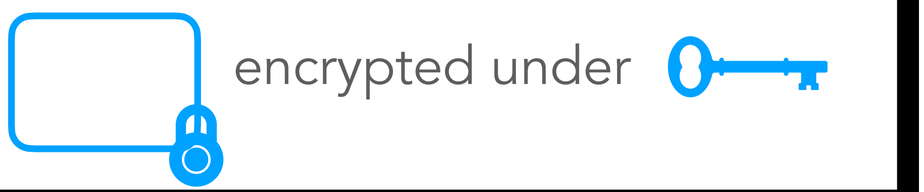

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Programmable Bootstrapping (PBS)




≈ **Homomorphic Decryption**

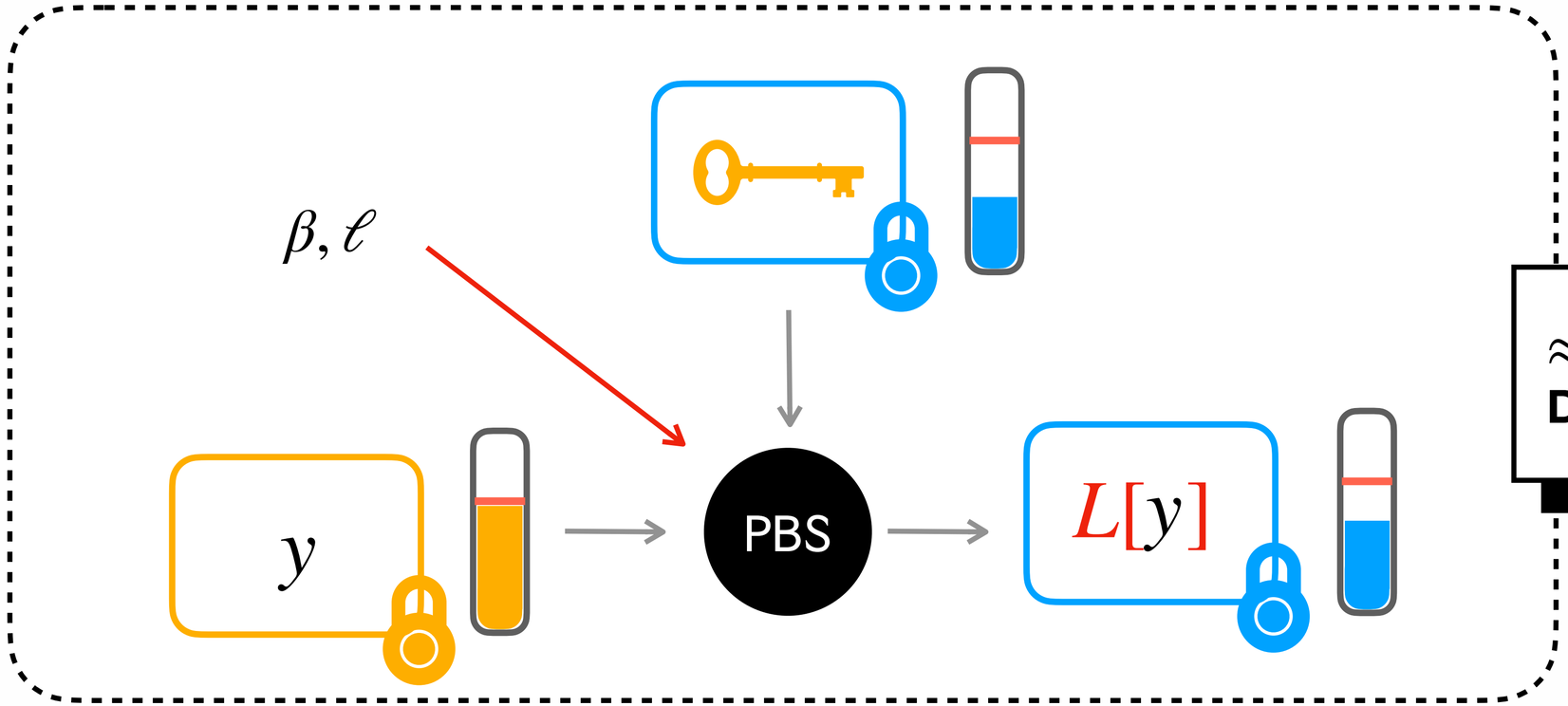


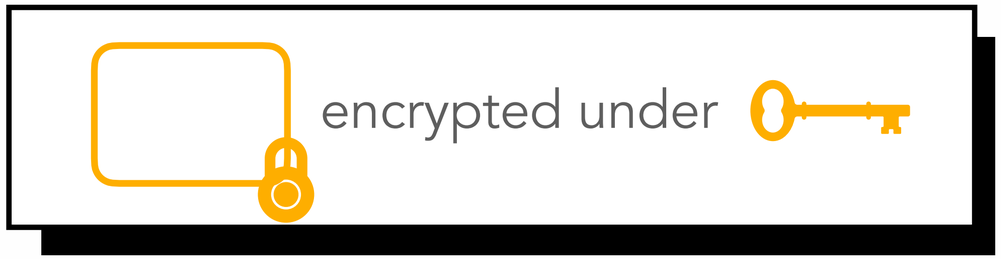


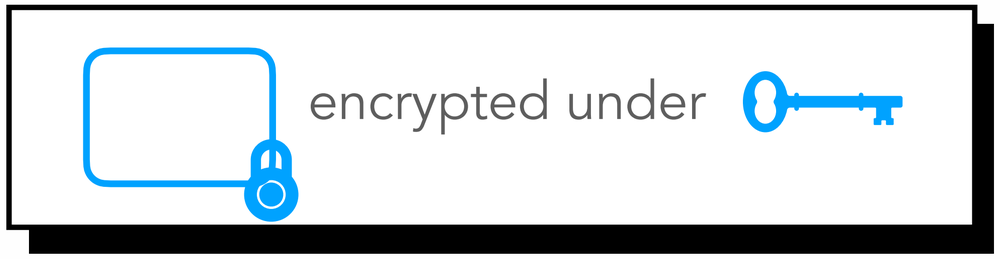

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Programmable Bootstrapping (PBS)




≈ **Homomorphic Decryption**

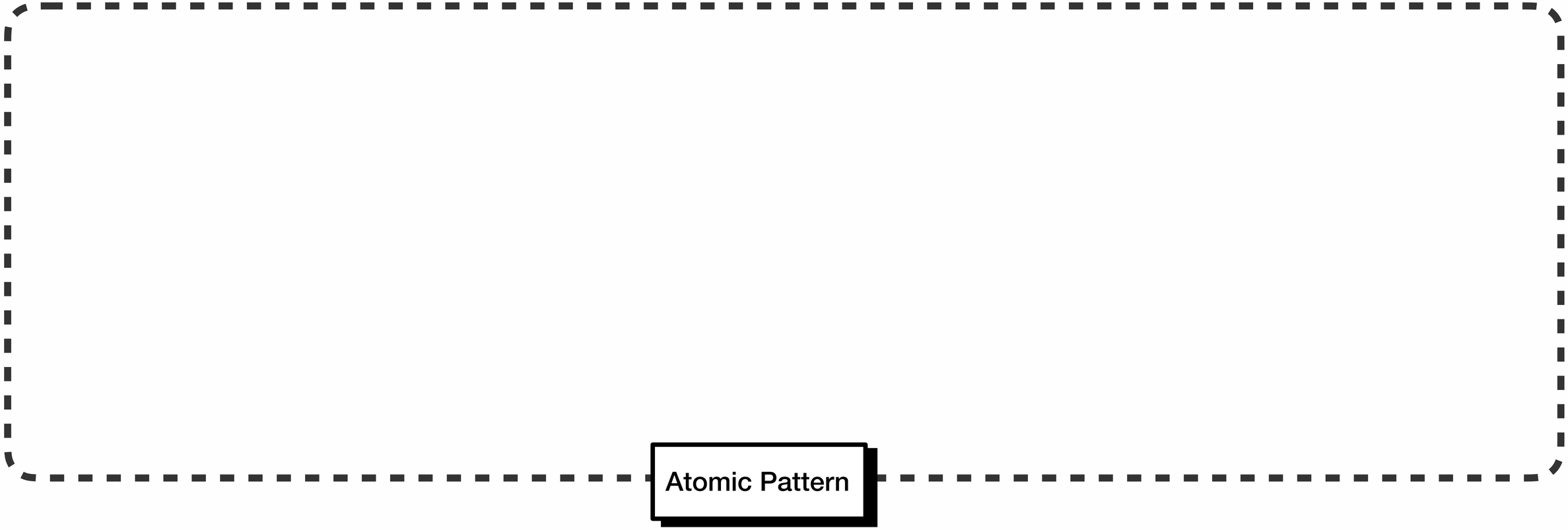




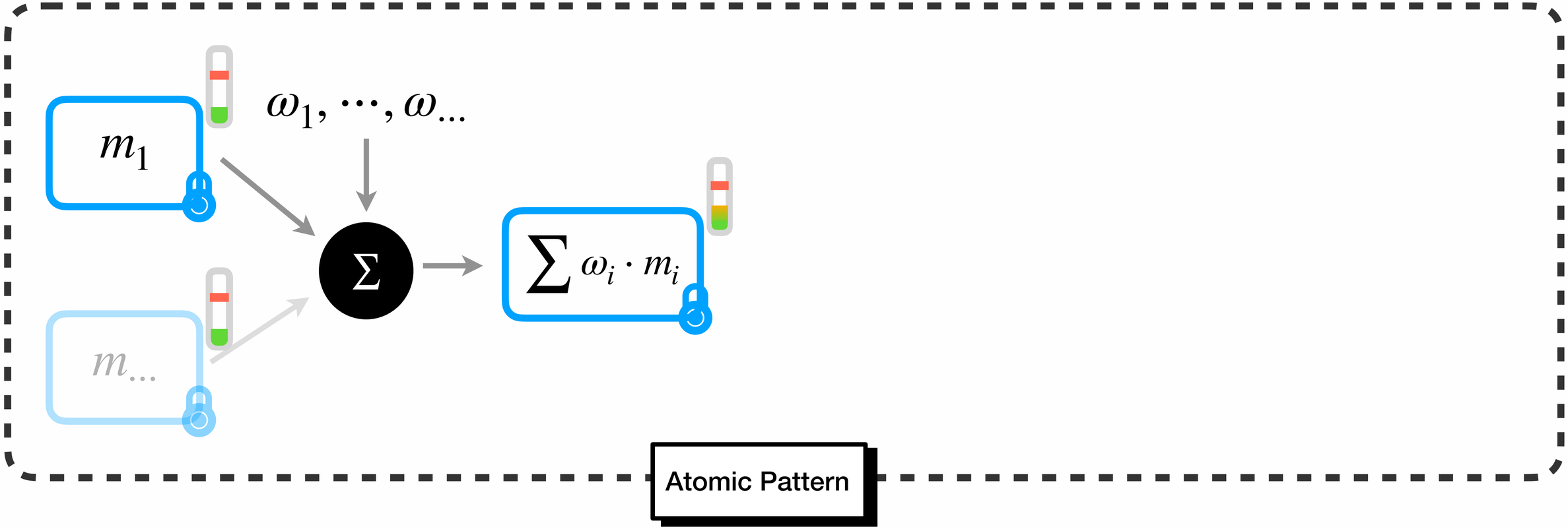

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.

# Optimal arrangement of TFHE building blocks



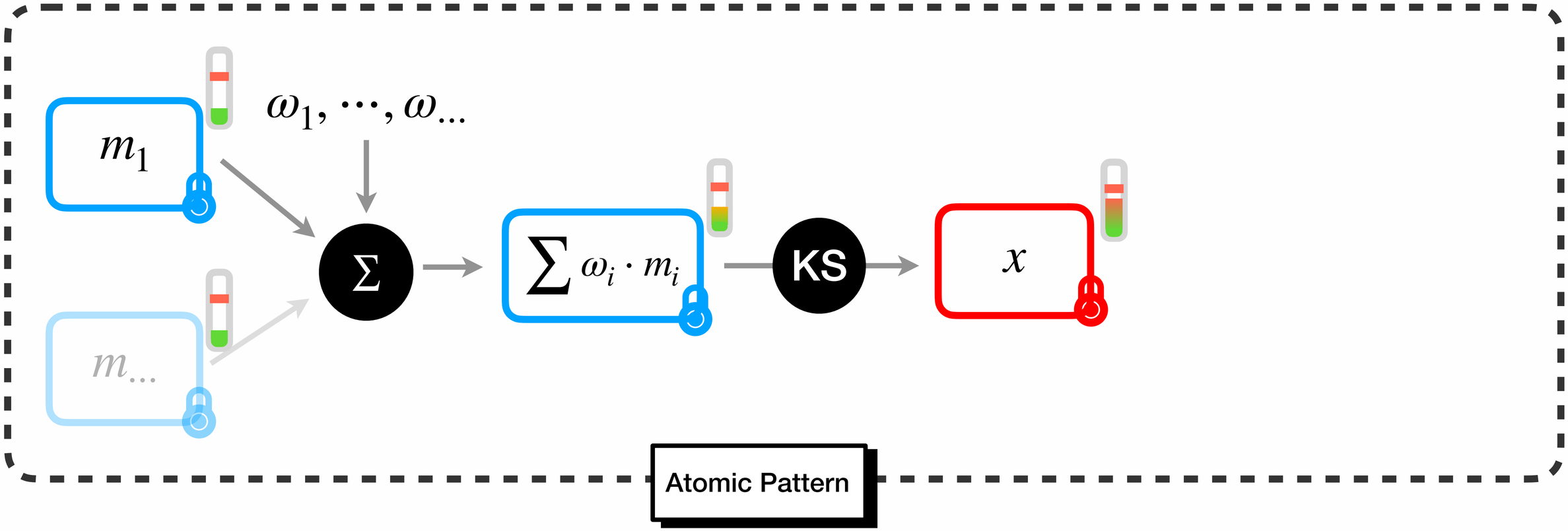

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. To appear in JoC

# Optimal arrangement of TFHE building blocks



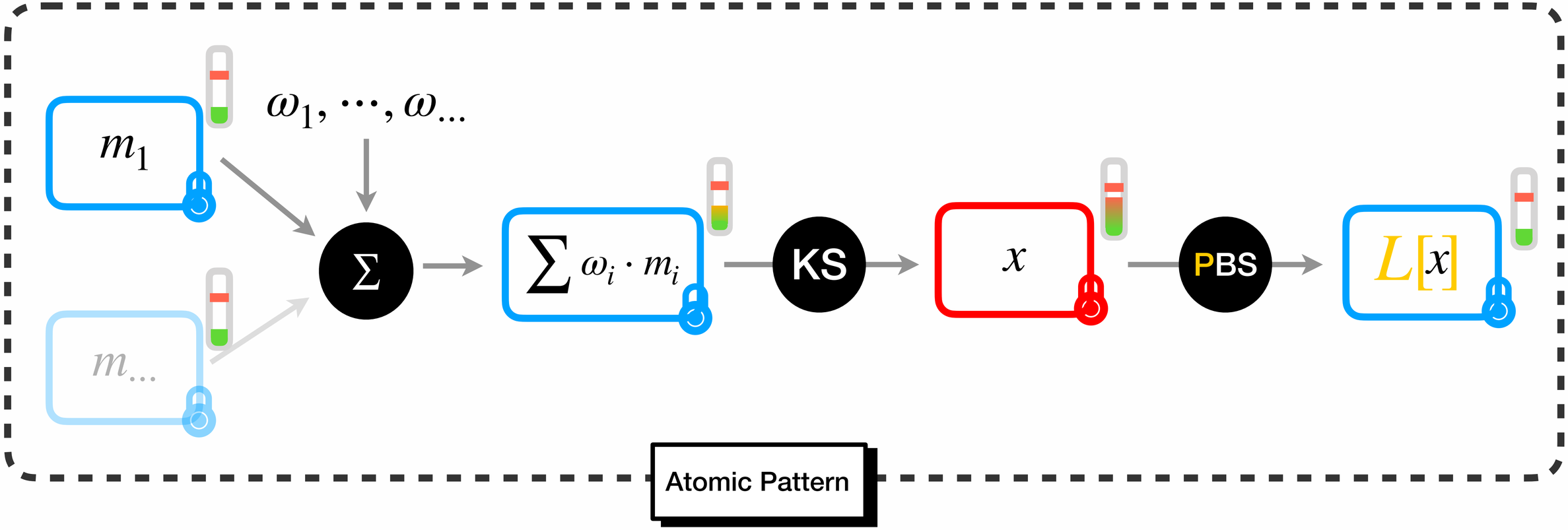

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. To appear in JoC

# Optimal arrangement of TFHE building blocks




**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. To appear in JoC

# Optimal arrangement of TFHE building blocks




**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. To appear in JoC

# Concrete Optimizer

An optimizer for TFHE

# Goals

Translate a function $f$ into a graph of TFHE operators

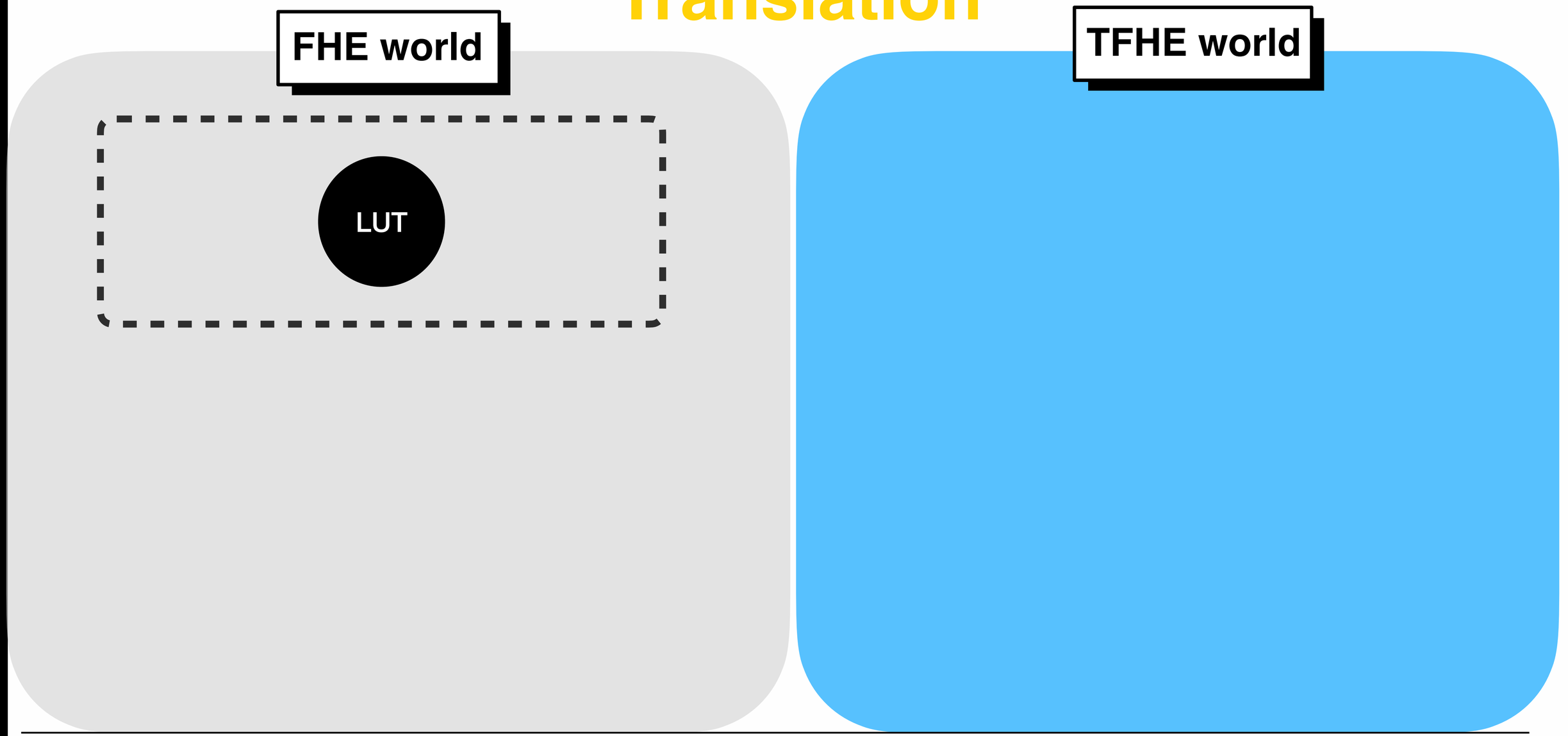
# Translation

**FHE world**

**TFHE world**

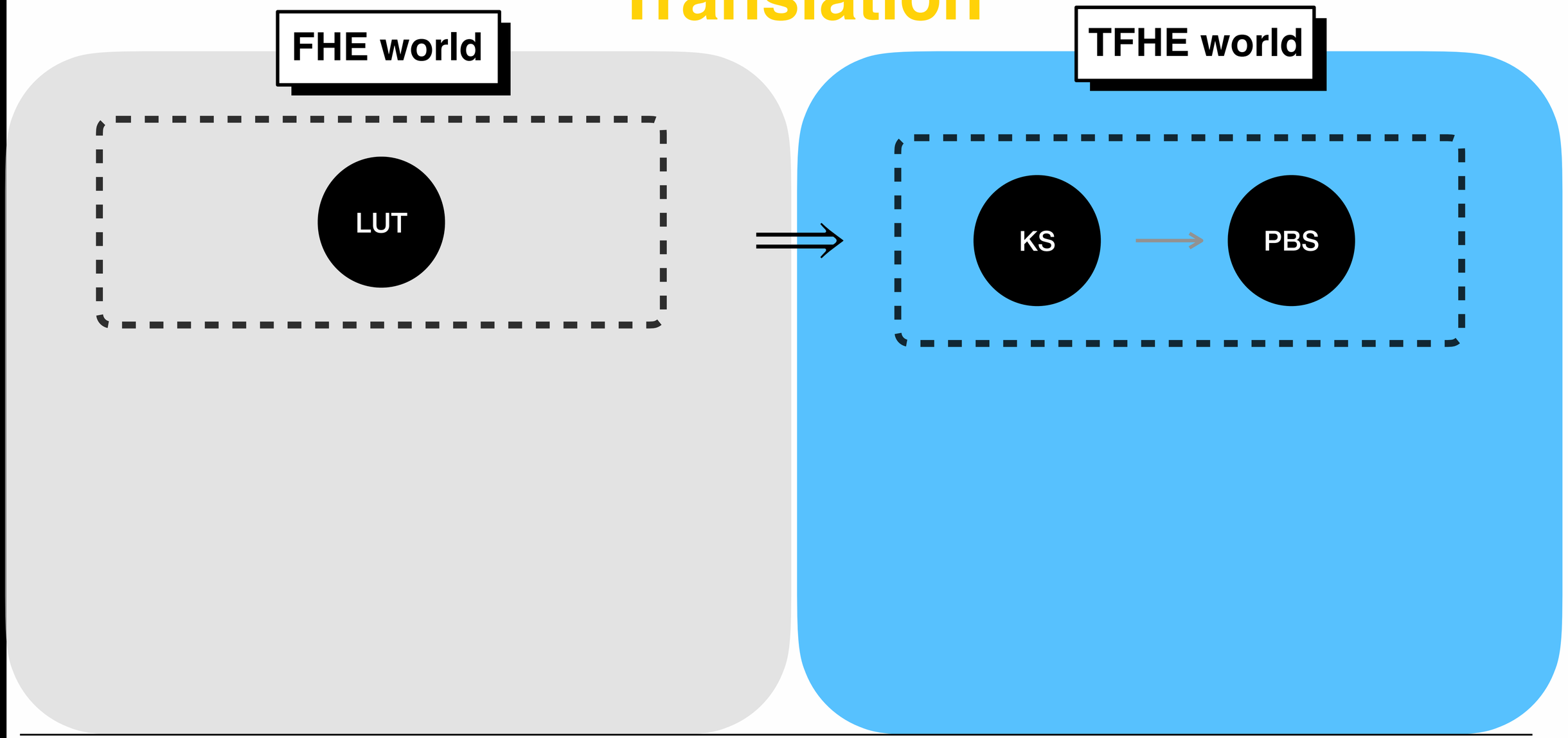
**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



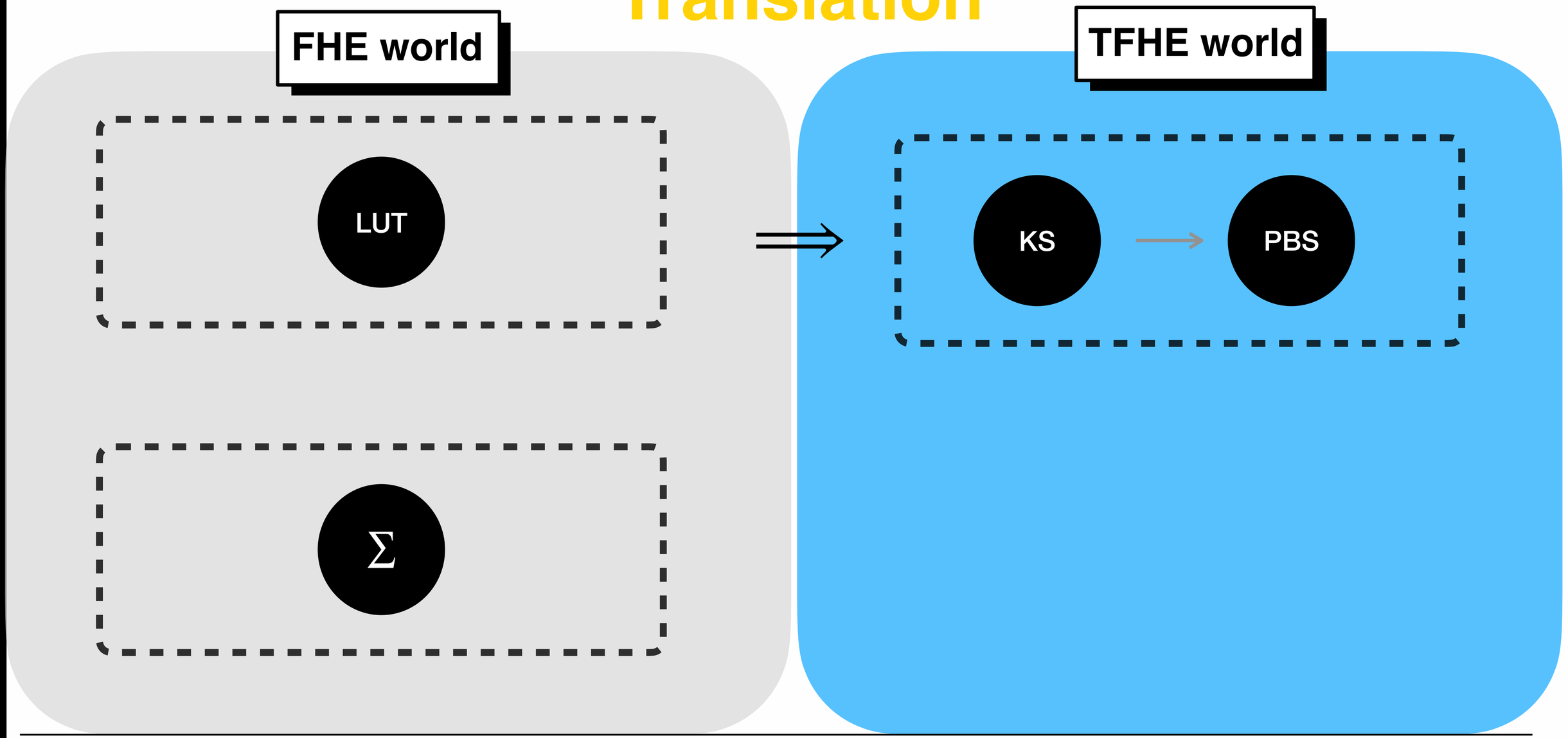

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



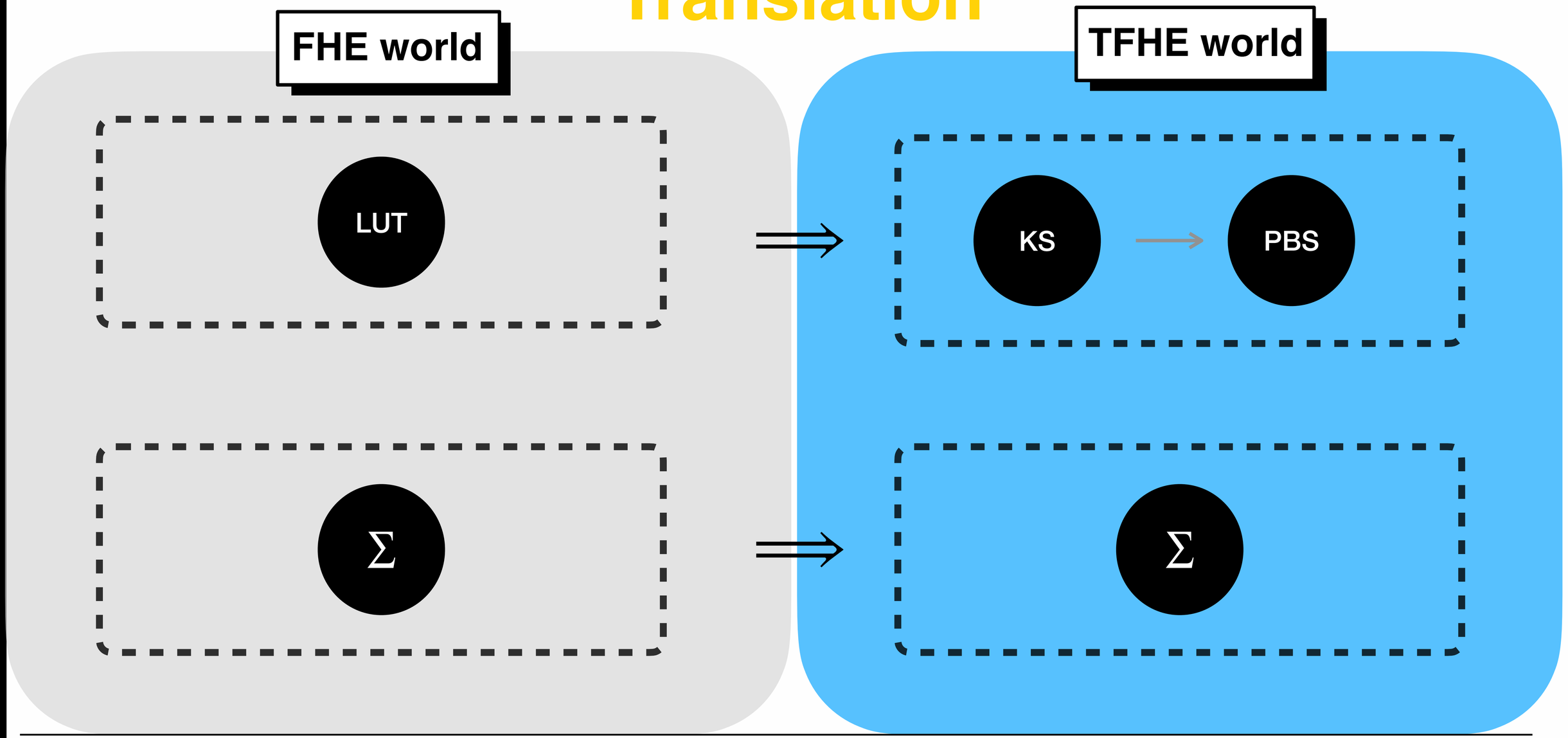

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



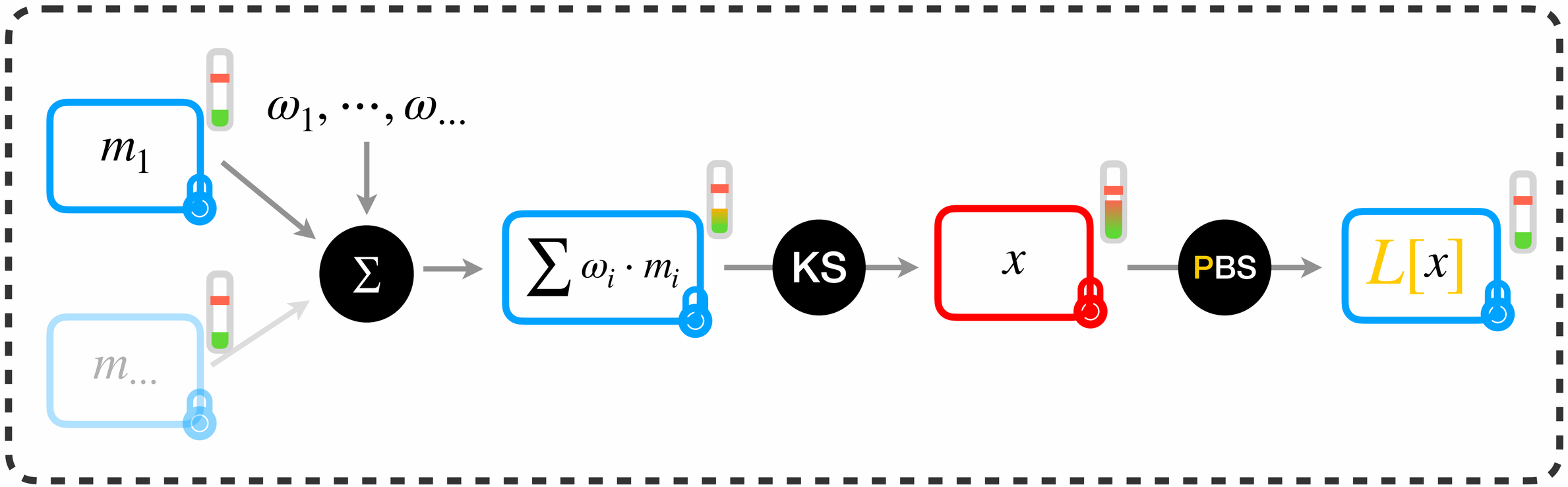

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



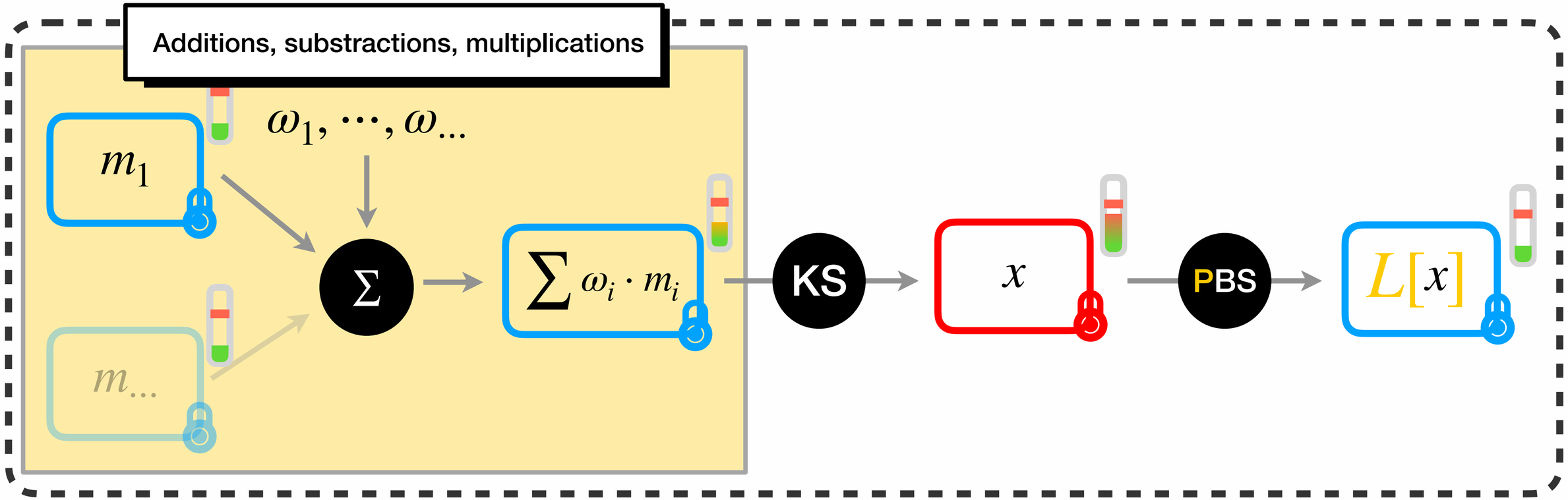

**[CGGI20]** I. Chillotti, N. Gama, M. Georgieva, M. Izabachène. TFHE: Fast Fully Homomorphic Encryption over the Torus. Journal of Cryptology 2020.
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



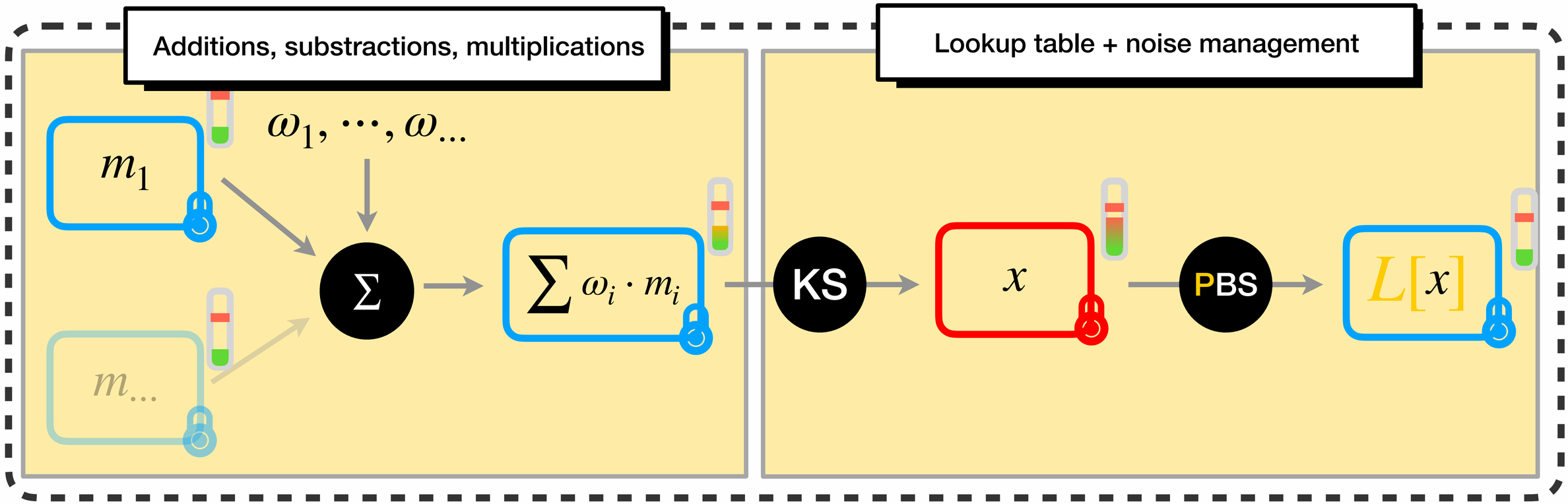

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



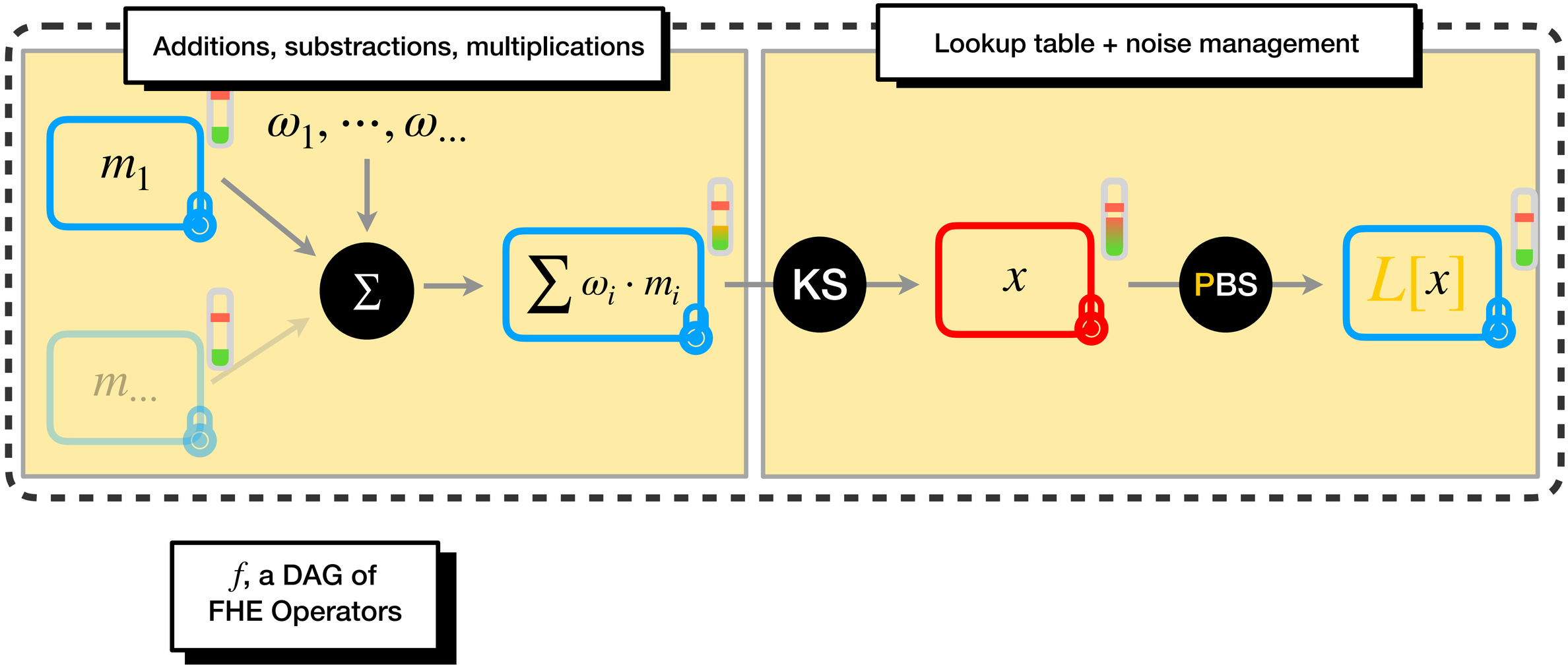

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



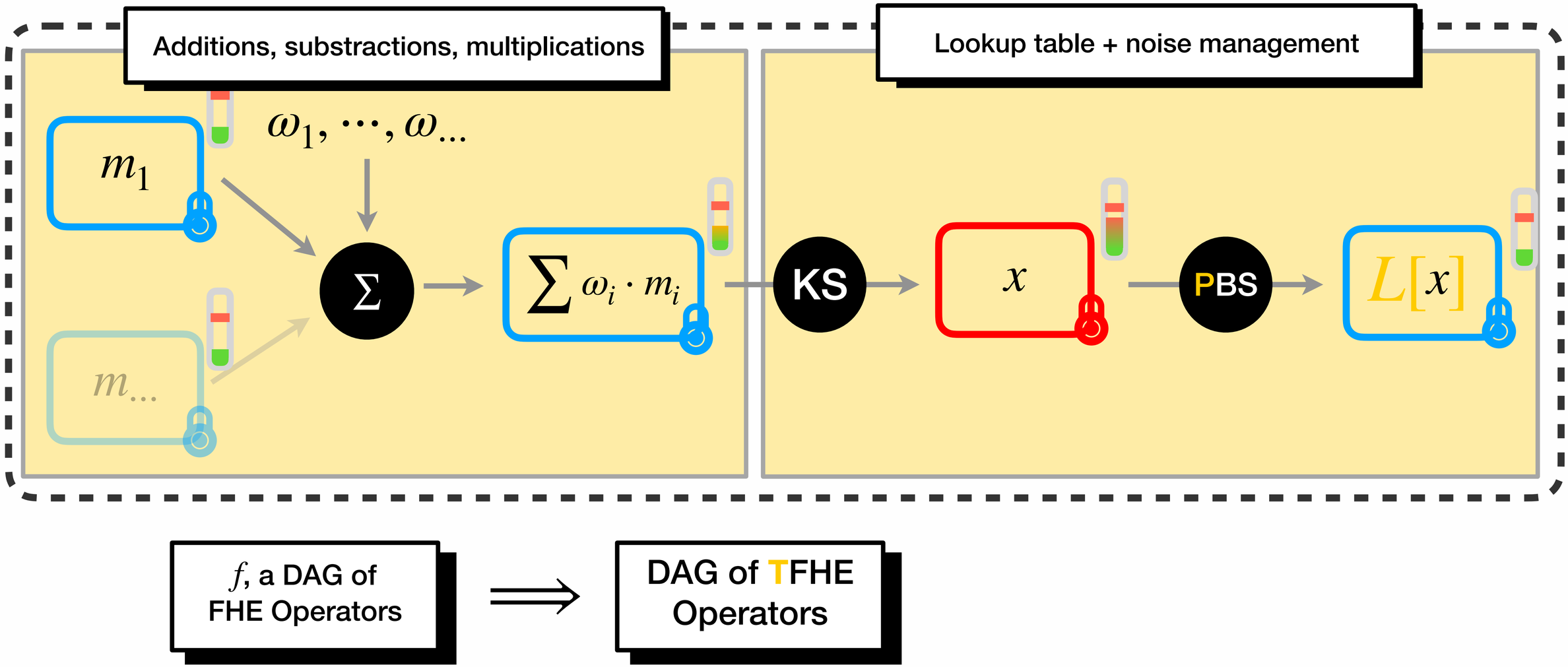

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



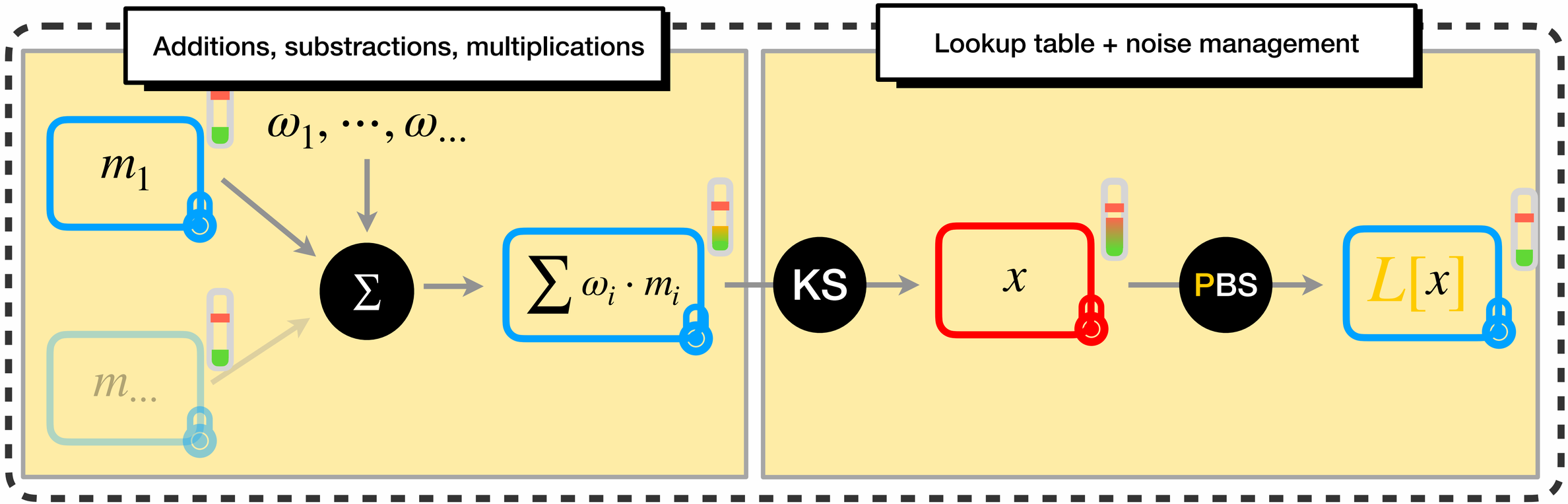

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation




**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

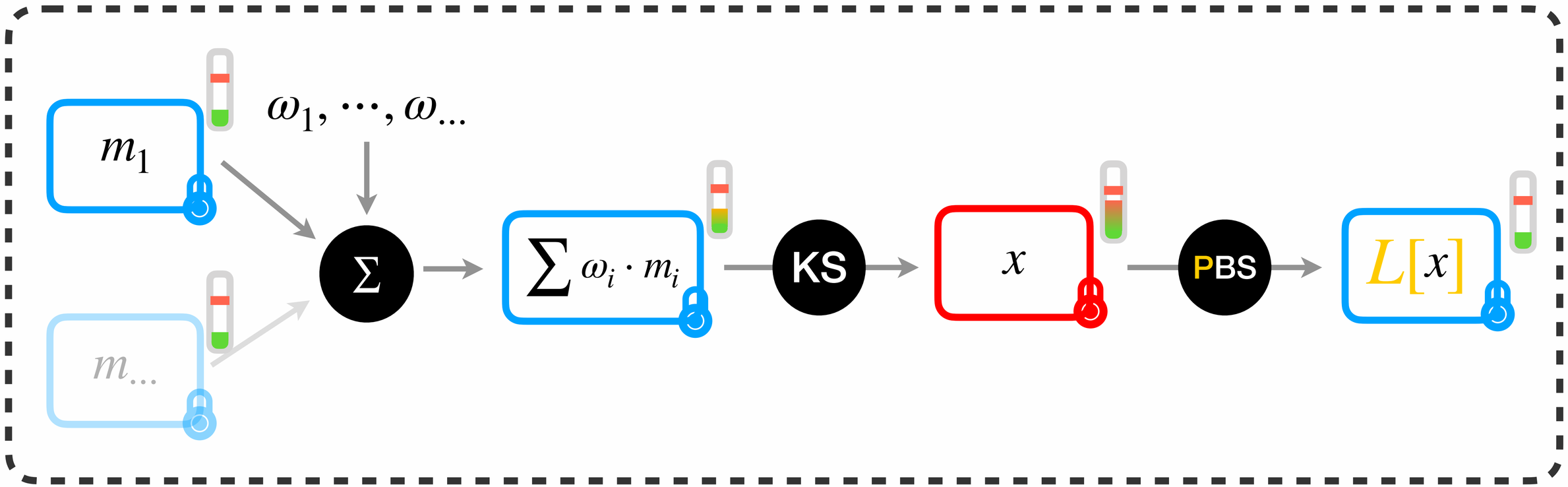
# Translation




$f$, a DAG of FHE Operators $\Longrightarrow$ DAG of TFHE Operators $\Longrightarrow$ DAG of atomic patterns

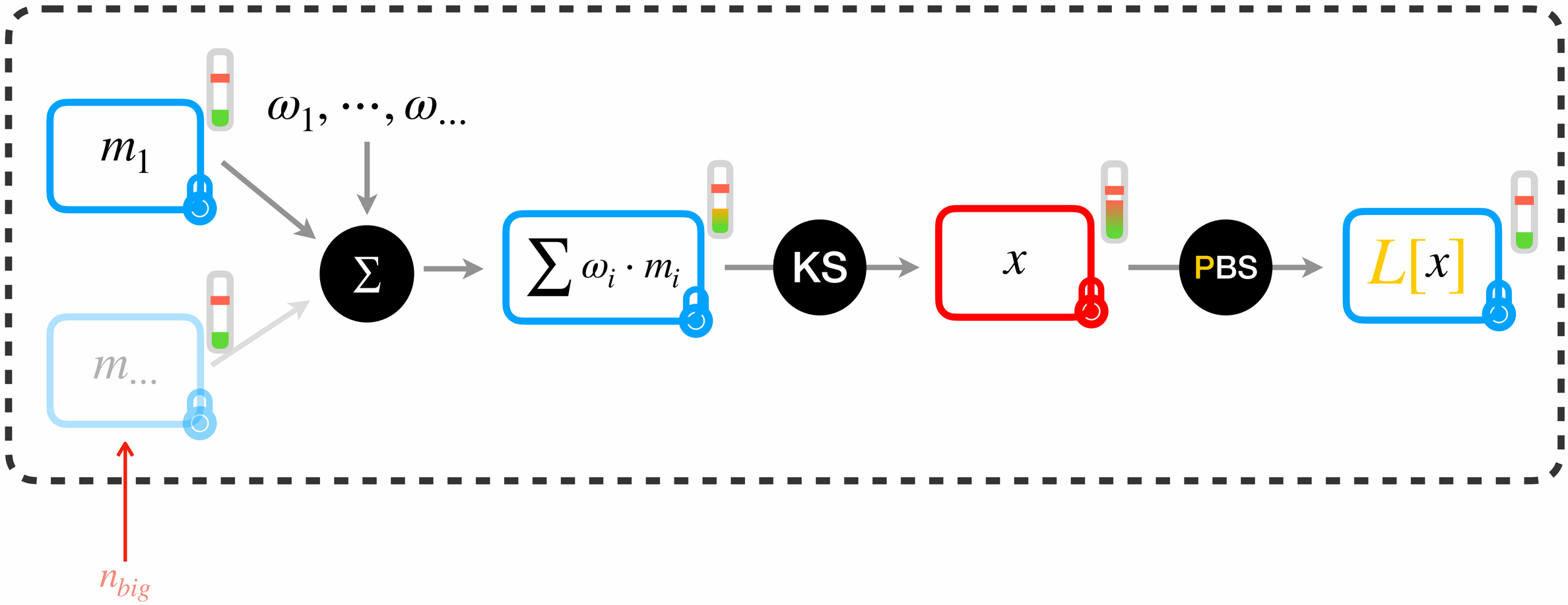
**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



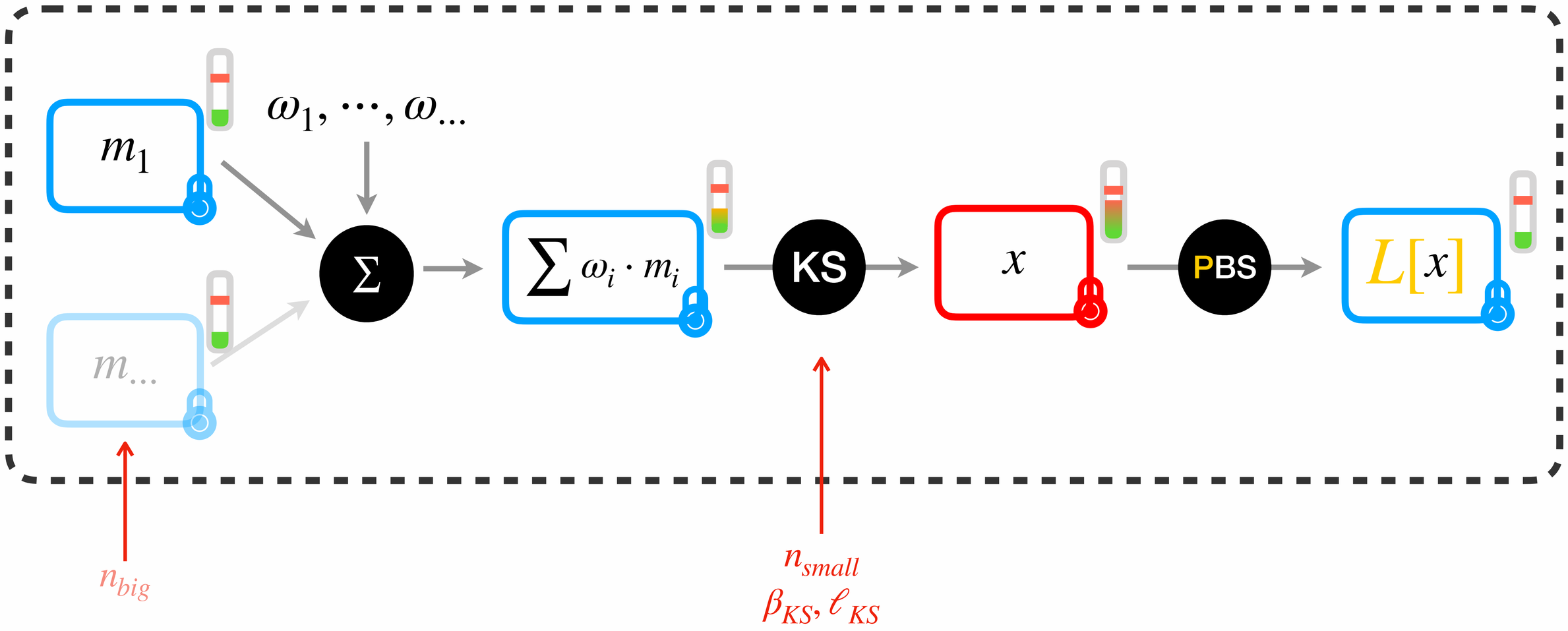

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



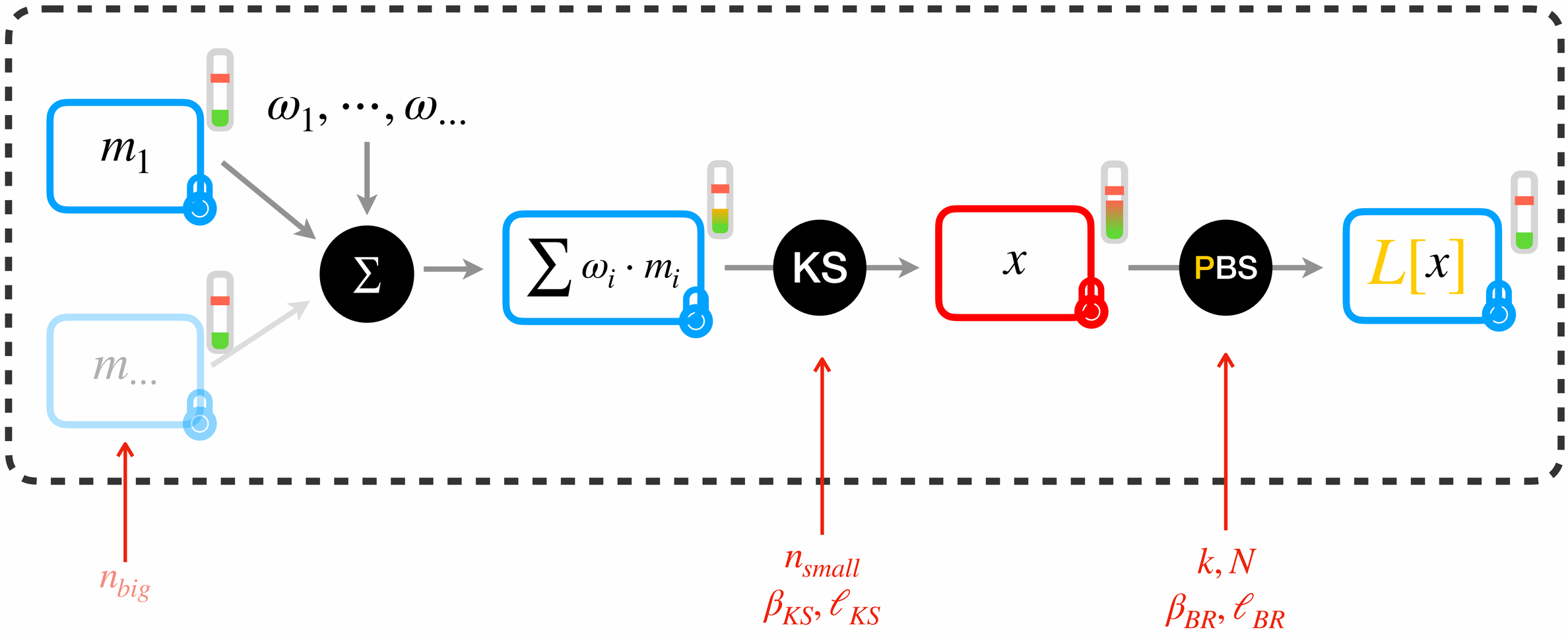

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation



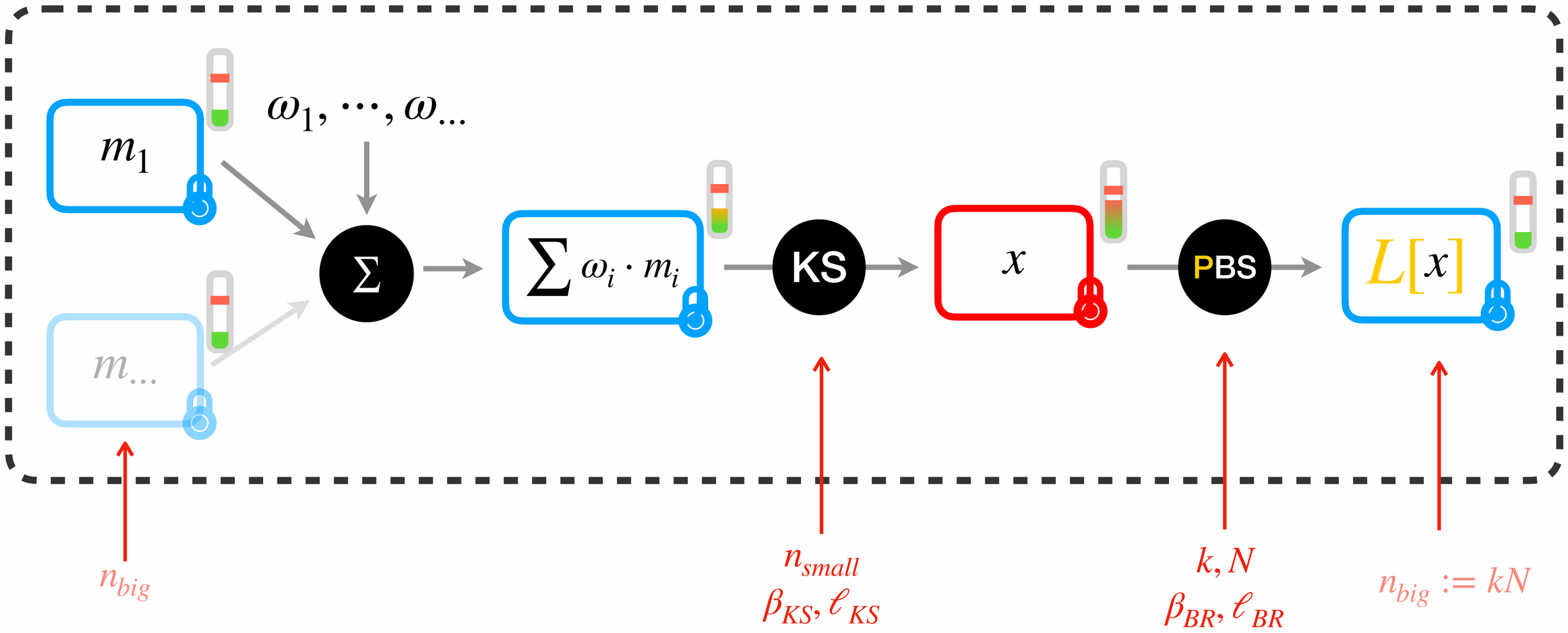

**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation




**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Translation




**[CJP21]** I. Chillotti, M. Joye, and P. Paillier. Programmable bootstrapping enables efficient homomorphic inference of deep neural networks. In CSCML 2021
**[BBB+22]** L. Bergerat, A. Boudi, Q. Bourgerie, I. Chillotti, D. Ligier, J.-B. Orfila, S. Tap Parameter Optimization & Larger Precision for (T)FHE. Eprint

# Goals

Translate a function $f$ into a graph of TFHE operators

with



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Goals

Translate a function $f$ into a graph of TFHE operators

with

Security



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Goals

Translate a function $f$ into a graph of TFHE operators

with

Security

Correctness



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Goals

Translate a function $f$ into a graph of TFHE operators

with

Security

Correctness

Efficiency



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Security 



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Security 

## Lattice Estimator [APS]

Reference tool to estimate the security of lattice-based encryption scheme **[APS]**

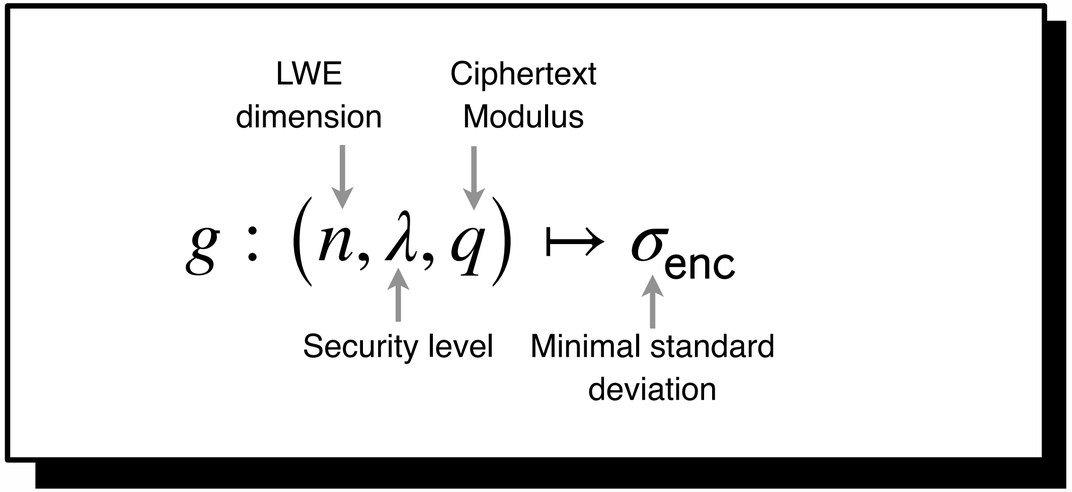


**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Security 

## Lattice Estimator [APS]

Reference tool to estimate the security of lattice-based encryption scheme **[APS]**



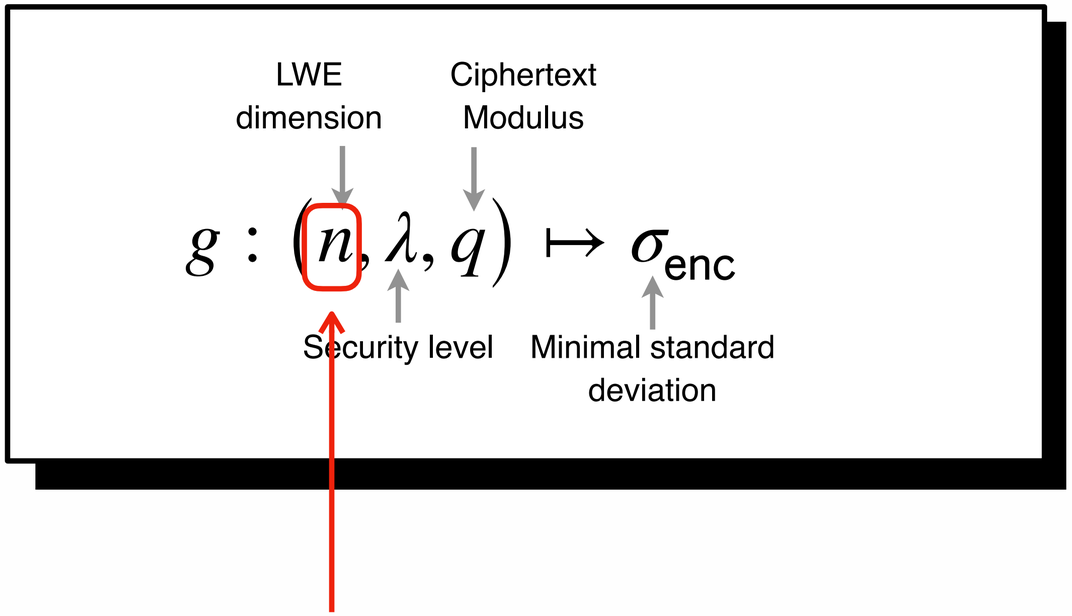



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Security

## Lattice Estimator [APS]

Reference tool to estimate the security of lattice-based encryption scheme **[APS]**



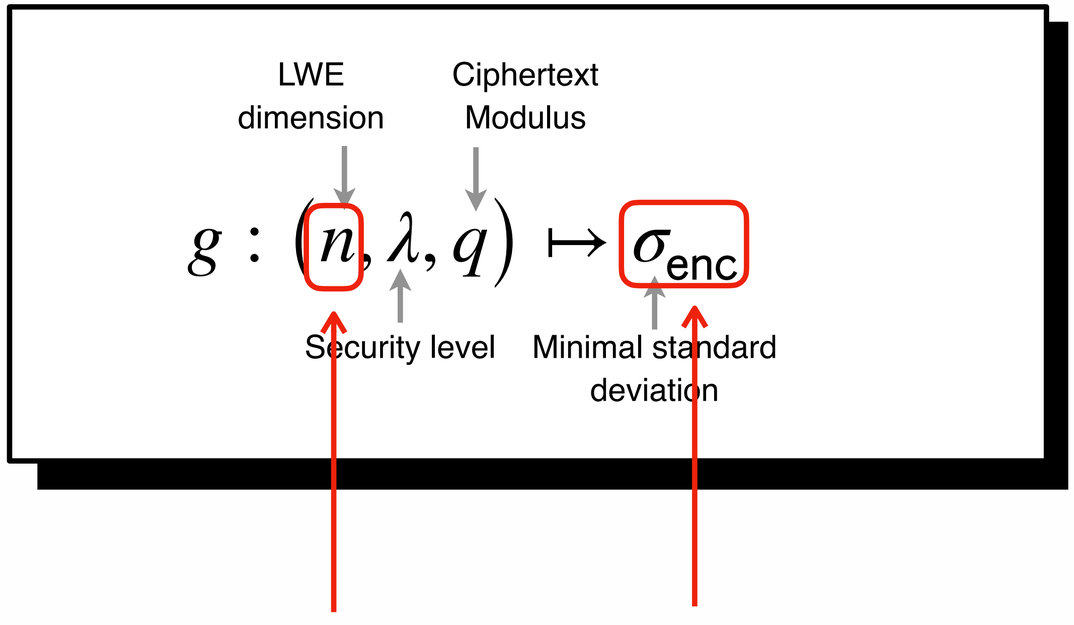



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Security

## Lattice Estimator [APS]

Reference tool to estimate the security of lattice-based encryption scheme **[APS]**






**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Efficiency 🚀



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Efficiency 🚀

**Rule of thumb**

The smaller the parameters, the faster the computation



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Efficiency 🚀

**Rule of thumb**

The smaller the parameters, the faster the computation

🤔 **How can we rank the feasible parameters ?**



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

# Efficiency 🚀

**Rule of thumb**

The smaller the parameters, the faster the computation

🤔 **How can we rank the feasible parameters ?**

$\Longrightarrow$

**Cost Model** as a surrogate of the execution time

*Ex: algorithmic complexities*



**[APS]** M. R. Albrecht, R. Player and S. Scott. On the concrete hardness of Learning with Errors. Journal of Mathematical Cryptology 2015

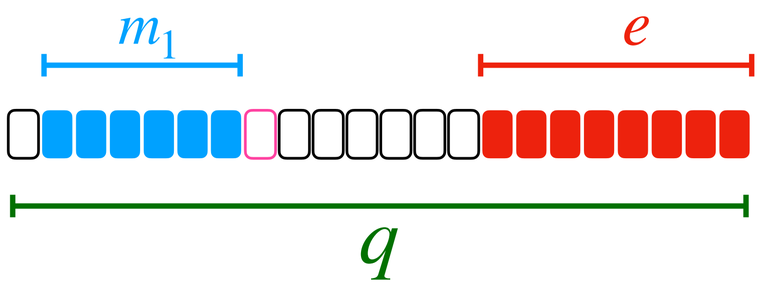
# Correctness ✅

# Correctness ✅

# Correctness ✅

**Correctness Condition**

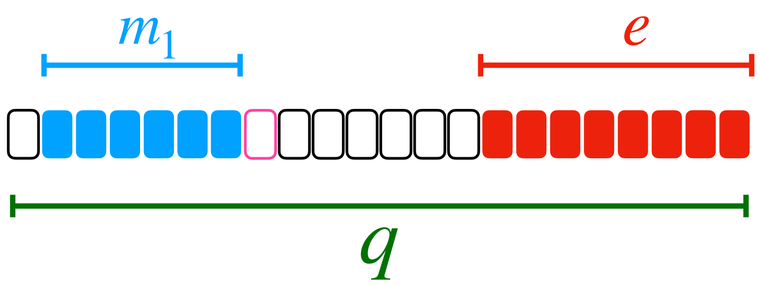
$$|e| < \frac{\Delta}{2}$$



# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2}$$

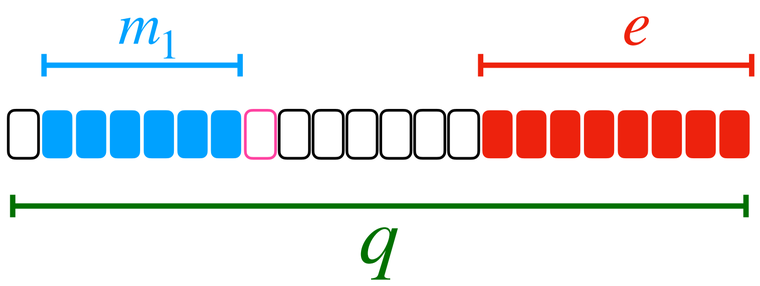



⚠️ $e$ can not be publicly known without compromising the security

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2}$$



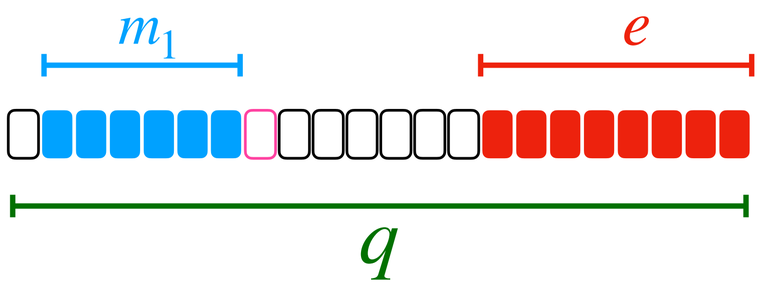

⚠️ $e$ can not be publicly known without compromising the security

💡 The distribution of $e$ can be publicly known without compromising the security

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2}$$

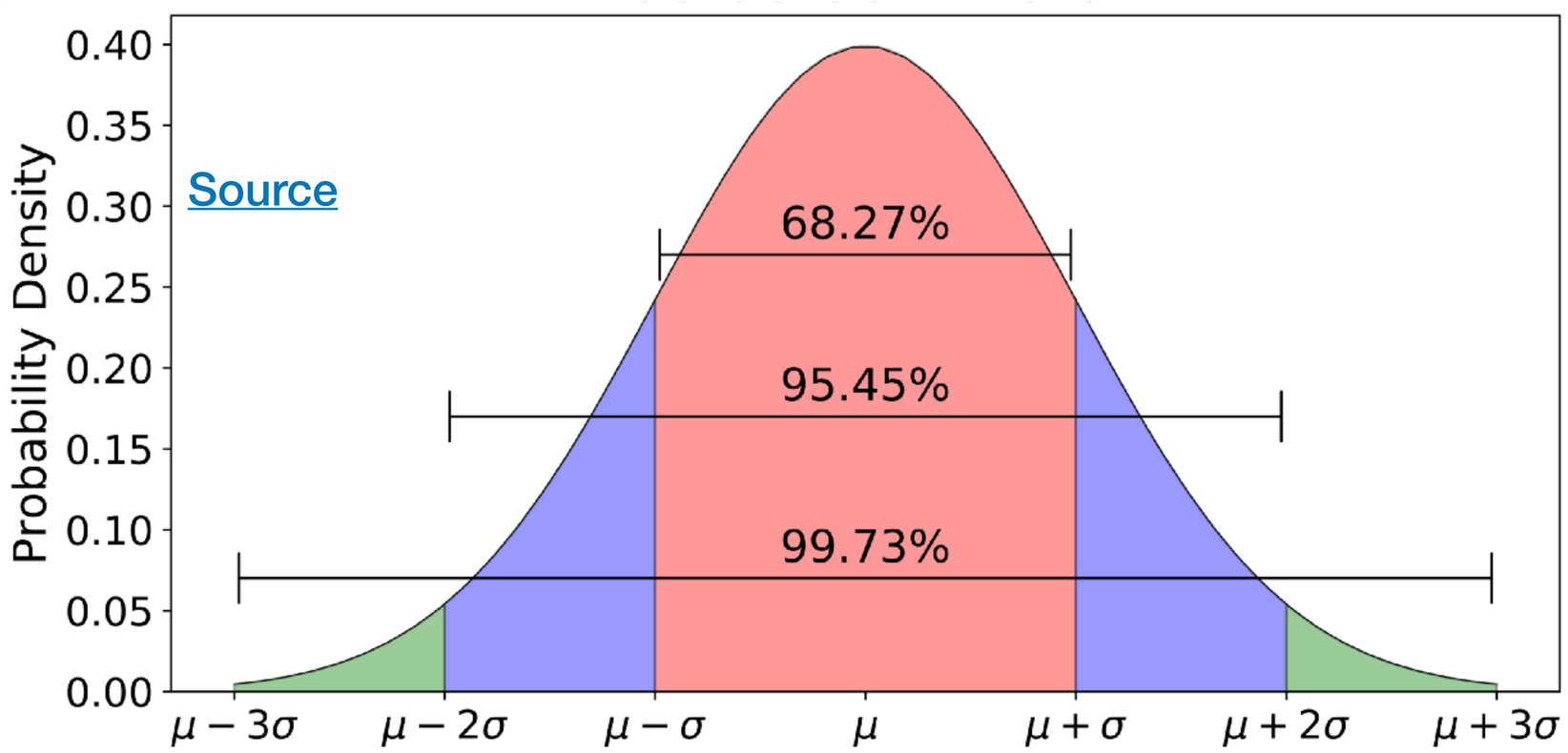



⚠️ $e$ can not be publicly known without compromising the security

💡 The distribution of $e$ can be publicly known without compromising the security

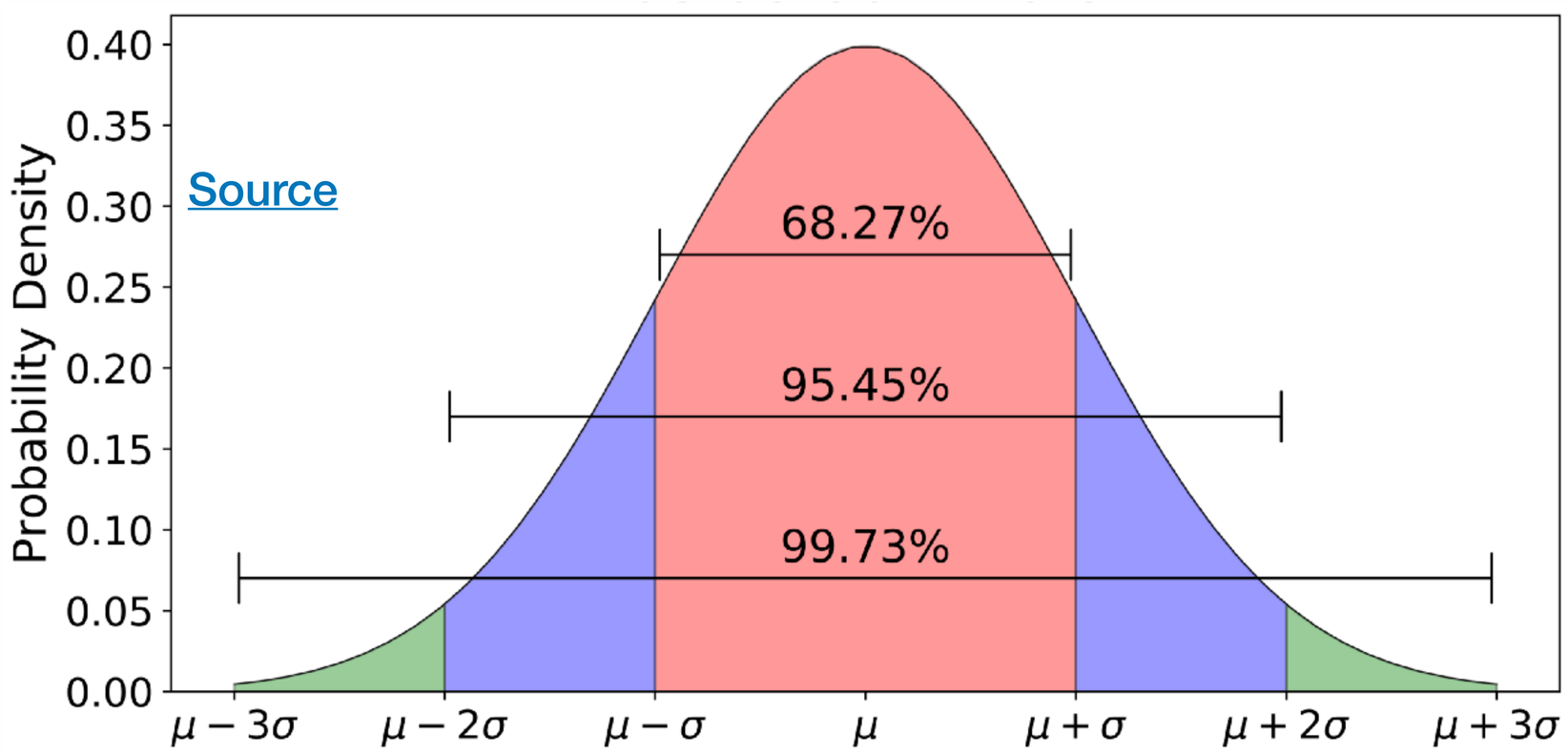
$\Longrightarrow$ **Noise Model** to track the noise distribution throughout the computation

# Correctness ✅




e is drawn from $\mathcal{N}\left(\mu = 0, \sigma^2\right)$

# Correctness ✅




e is drawn from $\mathcal{N}\left(\mu = 0, \sigma^2\right)$ $\Longrightarrow$ $\Pr\left[e \notin [-\kappa \cdot \sigma, \kappa \cdot \sigma]\right] = p_{\text{fail}}$

# Correctness ✅

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2}$$

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability } 1 - p_{\text{fail}}$$

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability } 1 - p_{\text{fail}}$$

$\Leftrightarrow$

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability} \quad 1 - p_{\text{fail}}$$

$$\Leftrightarrow$$

$$\sigma \leq \frac{\Delta}{2 \cdot \kappa}$$

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability} \quad 1 - p_{\text{fail}}$$

Noise (private)

$$\Leftrightarrow$$

$$\sigma \leq \frac{\Delta}{2 \cdot \kappa}$$

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability} \quad 1 - p_{\text{fail}}$$

Noise (private)

$$\Leftrightarrow$$

$$\sigma \leq \frac{\Delta}{2 \cdot \kappa}$$

Noise Distribution (public)

# Correctness ✅

**Correctness Condition**

$$|e| < \frac{\Delta}{2} \quad \text{with probability } 1 - p_{\text{fail}}$$

Noise (private)

$\Leftrightarrow$

Noise bound

$$\sigma \leq \frac{\Delta}{2 \cdot \kappa}$$

Noise Distribution (public)

# Optimization Problem

# Optimization Problem

$$\min \quad \text{Cost} \quad \mathcal{G}$$

# Optimization Problem

$$\min \quad \text{Cost} \quad \mathscr{G} \quad \text{s.t.} \begin{cases} \forall i \in I, \sigma_i \leq \dfrac{\Delta}{2 \cdot \kappa} \end{cases}$$

# Optimization Problem

$$\min \quad \text{Cost}_{\mathscr{G}} \quad \text{s.t.} \begin{cases} \forall i \in I, \sigma_i \leq \dfrac{\Delta}{2 \cdot \kappa} \\ \sigma_{\text{enc}} = g\left(n, \lambda, q\right) \end{cases}$$

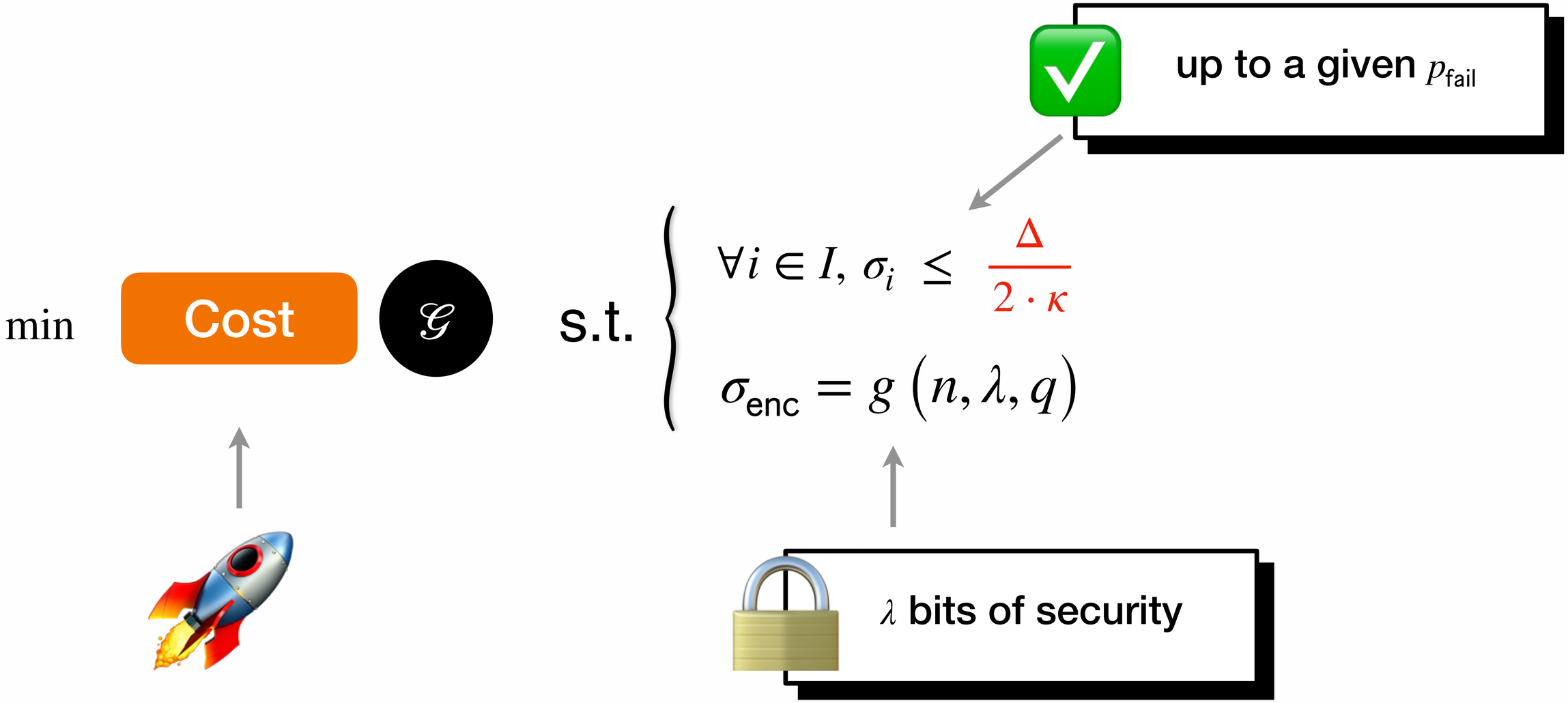
# Optimization Problem

$$\min \quad \text{Cost} \quad \mathcal{G} \qquad \text{s.t.} \begin{cases} \forall i \in I,\ \sigma_i \leq \dfrac{\Delta}{2 \cdot \kappa} \\ \sigma_{\text{enc}} = g\left(n, \lambda, q\right) \end{cases}$$

# Optimization Problem


$$\min \quad \text{Cost} \quad \mathcal{G} \qquad \text{s.t.} \begin{cases} \forall i \in I, \sigma_i \leq \dfrac{\Delta}{2 \cdot \kappa} \\ \sigma_{\text{enc}} = g\left(n, \lambda, q\right) \end{cases}$$

# Optimization Problem



# Conclusion

# What's next ?

## Translation

More than one possible translation: how to find the best one ?

## Multi Parameter Set

One parameter set vs several parameter set

## Failure Probability

At operator level vs at graph level

## Optimization Problem

Simplification with domination reasoning

# Part II

How to translate an arbitrary DAG into an FHE DAG i.e., a DAG only composed of

linear operations and 

?

How to convert a TFHE DAG into a binary executable ?

# Contact and Links

samuel.tap@zama.ai

zama.ai

github.com/zama-ai/concrete

community.zama.ai

ZAMA