MONET +

Post Quantum Challenge

business breakfast

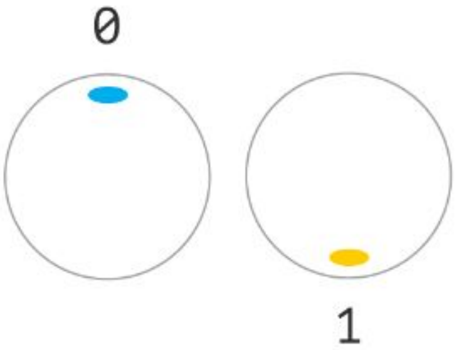
Mgr. Anežka Pejlová
Security Architect & RECS team lead

What is PQC?

Classical vs. Quantum computer

bits

- 0 or 1
- nothing in between
- classical

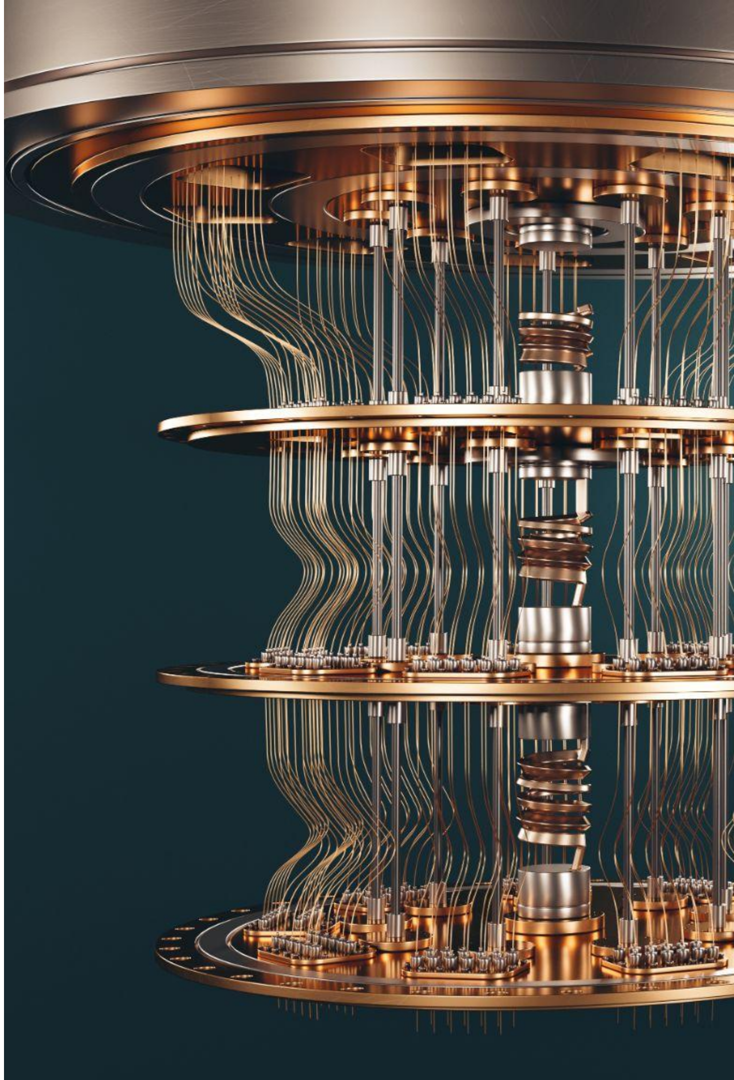


qubits

- any superposition between 0 and 1
- measurement = final state
- dependent on the probability of superposition
- quantum

Quantum computers

- change of paradigm in computer world
- more effective solution to some hard problems
- significant progress/breakthrough in
  - AI
  - optimization problems
  - discovery/development process
  - financial modeling
  - weather forecasting
  - cybersecurity
  - …



Quantum impact on classical cryptography

Shor's algorithm (1994)

- factorization (RSA)
- discrete logarithm (DH, ECC)

Asymmetric cryptography

Grover's algorithm (1996)

- state space search (keys, collisions)
- probably hard parallelizable

Symmetric cryptography

Post-quantum cryptography (PQC)

- **cryptography secure against attacks by CRQC**
- based on “hard” mathematical problems from different areas:
  - error-correcting codes
  - lattices
  - hash functions
  - multivariate polynomials
  - isogeny of supersingular elliptic curves
- PQ algorithms are feasible on classical computers
  - vs. quantum cryptography (ie. QKD)

Why to bother with PQC?

Quantum impact on classical cryptography

Which systems are **NOT affected** by CRQC?

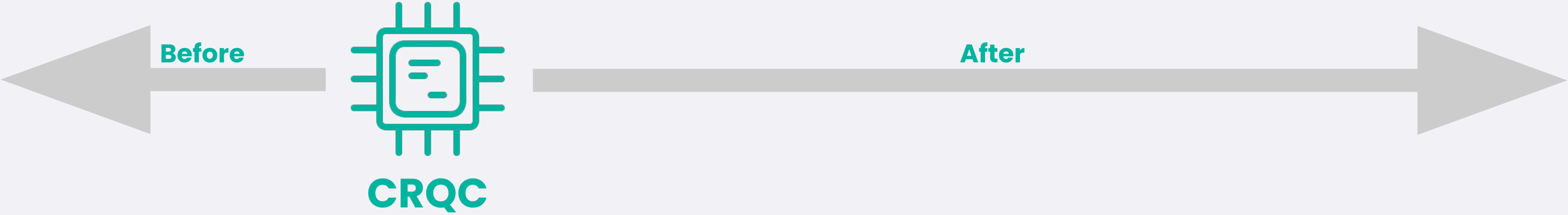
What does it mean “affected”?

Encryption

- confidentiality
- privacy

Signature

- integrity
- non-repudiation
- authentication



Store now…

…Decrypt later

Impersonate users by fraudulent authentication

Manipulate digitally signed documents / certificates

Are we ready?

Layered Matrix Challenge

Standards	Implementations	Adoption
protocols	SW/system/apps	solution
formats & structures	HW modules	infrastructure
approach	HW acceleration	organization
algorithms	libs	people

Standards

Algorithms

- **NIST** standards (08/2024)
    - ML-KEM
    - ML-DSA
    - SLH-DSA
    - FN-DSA (draft)
- **IETF RFCs** (2018/2019)
    - XMSS signatures
    - LM signatures
- **KpqC** (01/2025)
    - HATAE ~ ML-DSA, AIMer
    - SMAUG-T ~ ML-KEM, NTRU+

Coming standards

NIST

- FN-DSA (draft)
- additional KEMs → HQC (03/25)
- on-ramp signatures

China

- NGCC launch 02/2025
- PK, hash, block cipher expected

EU

- ISO/IEC 18033-2:2006/CD Amd 2 - under development
- incl. of NIST standards expected
- joint statement of 18 EU states

Standards for usage

Identifiers & formats

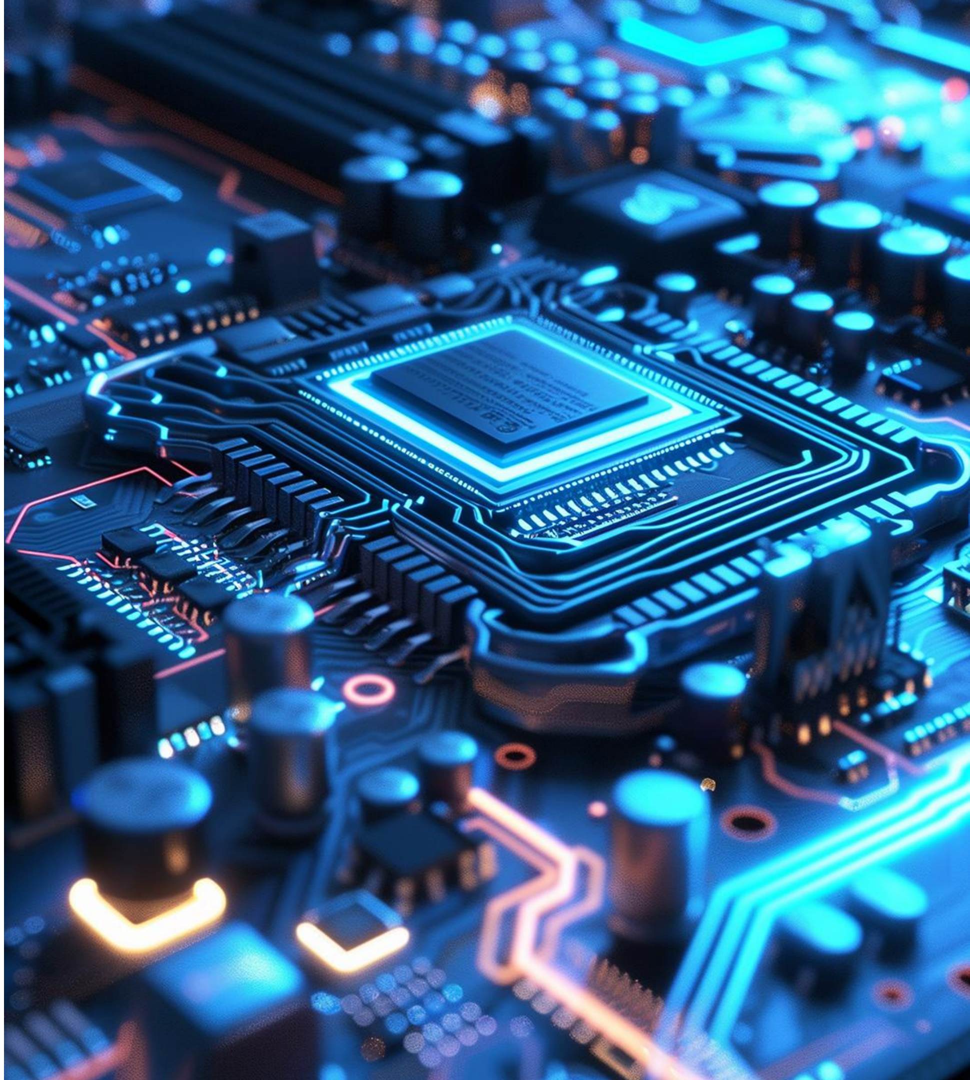
- OID
- NIST CSOR
- JOSE and COSE
- XML still missing

Usage

- ITU-T / ISO-IEC / RFC (X.509)
- TLS
- CMS
- JWE
- SAML
- PKCS#11

Support in HW/SW

- OQS project
  - TLS, SSH, X.509, CMS, S/MIME
  - Utimaco, Thales, Entrust, IBM, Cisco, Debian, SandboxAQ, …
- proprietary implementations
  - Microsoft (SymCrypt)
  - Google (Tink)
  - …
- HSMs and SCs
  - Thales
  - IBM, Entrust



Layered Matrix Challenge

Standards	Implementations	Adoption
protocols	SW/system/apps	solution
formats & structures	HW modules	infrastructure
approach	HW acceleration	organization
algorithms	libs	people

Cryptographic agility

Crypto-agility

- design supporting smooth change of crypto primitives without extensive system changes
- ideal - drop-in replacement
- without downtime of applications
- robust approach to implementing cryptography needed

Crypto-agility practically

- create **crypto-inventory** of all crypto assets and their dependencies
- develop **governance** and strategic roadmap on top of crypto-inventory
- design flexible and **modular architecture**
- ensure interoperability with use of **standardized interfaces/protocols**
- **avoid hardcoded** crypto algorithms and their parameters
- **automate** processes around PKI and key management

Crypto-agility practically

01

create **crypto-inventory** of all crypto assets and their dependencies

02

develop **governance** and strategic roadmap on top of crypto-inventory

03

design flexible and **modular architecture**

04

ensure interoperability with use of **standardized interfaces/protocols**

05

avoid hardcoded crypto algorithms and their parameters

06

automate processes around PKI and key management

Key takeaways

Mosca's Theorem

X = Security Shelf life

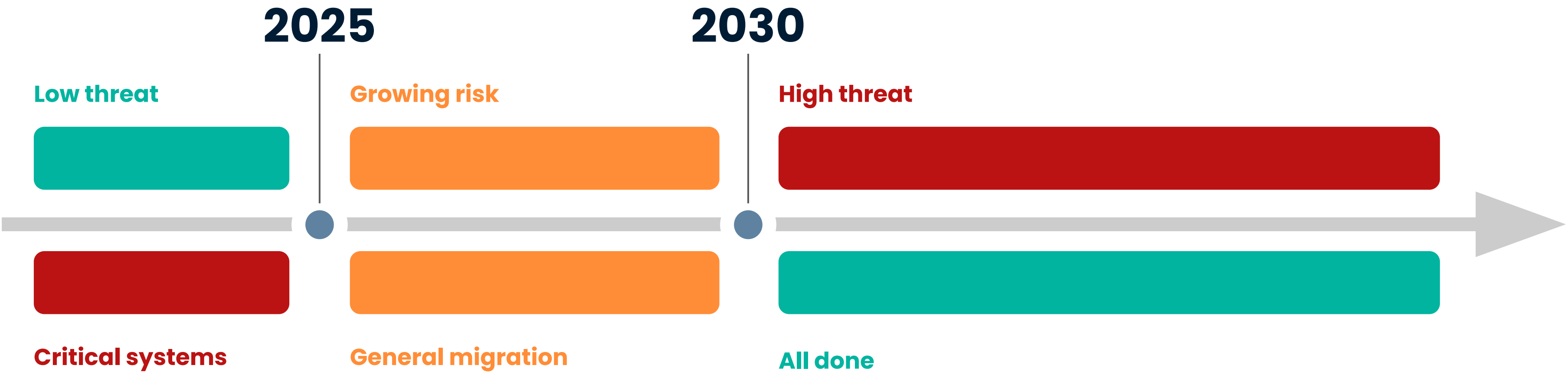
Y = Migration Time

Z = Time to compromise

If $X + Y > Z$ then system can be compromised!

CRQC maturity

Harvest now… …decrypt later



Start now… …relax later

PQC readiness

NOW!

Is the best time to start
with PQ migration preparation

How can we help you?

- map the environment
  - technical view
  - recommendation of (security) authorities
- create crypto-inventory
- build crypto-agile solutions
- define migration strategy for each case
- decide priorities
- prepare robust migration playbook
- migrate to PQ-ready solution case by case

MONET+

Thank you for your attention!

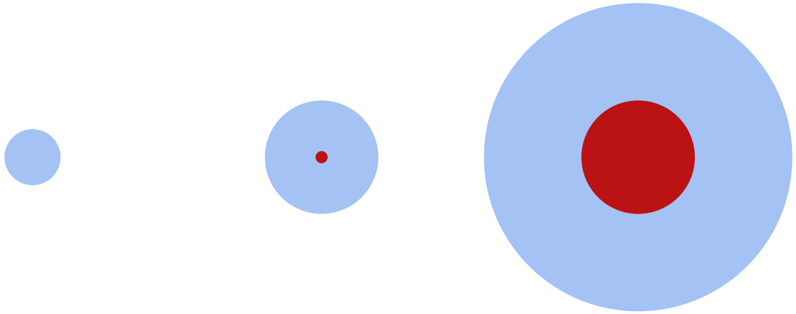
Contact me

Mgr. Anežka Pejlová
apejlova@monetplus.cz

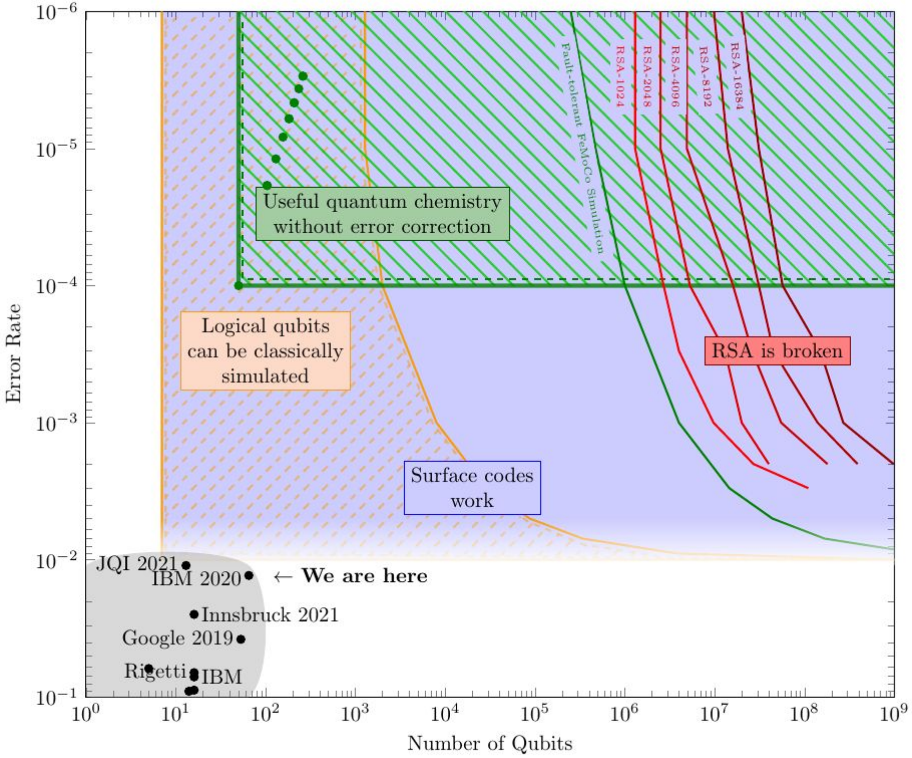
When?

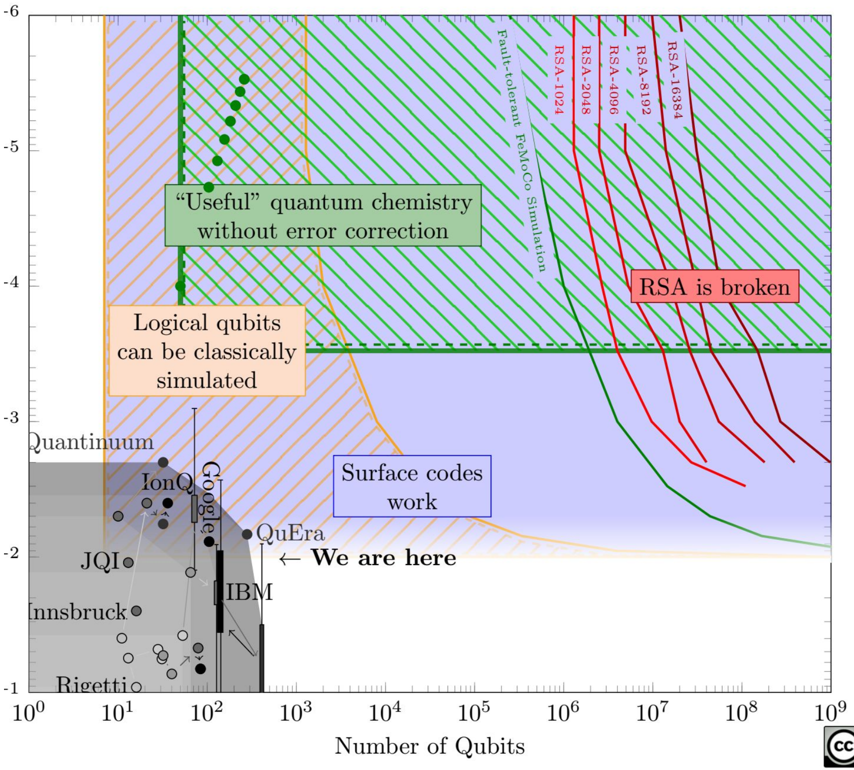
Evolution of quantum computers

→ 2018 → 2021 → 2024 → 2026 / 2031 / Later?



Landscape of Quantum Computing in 2021 vs. 2024

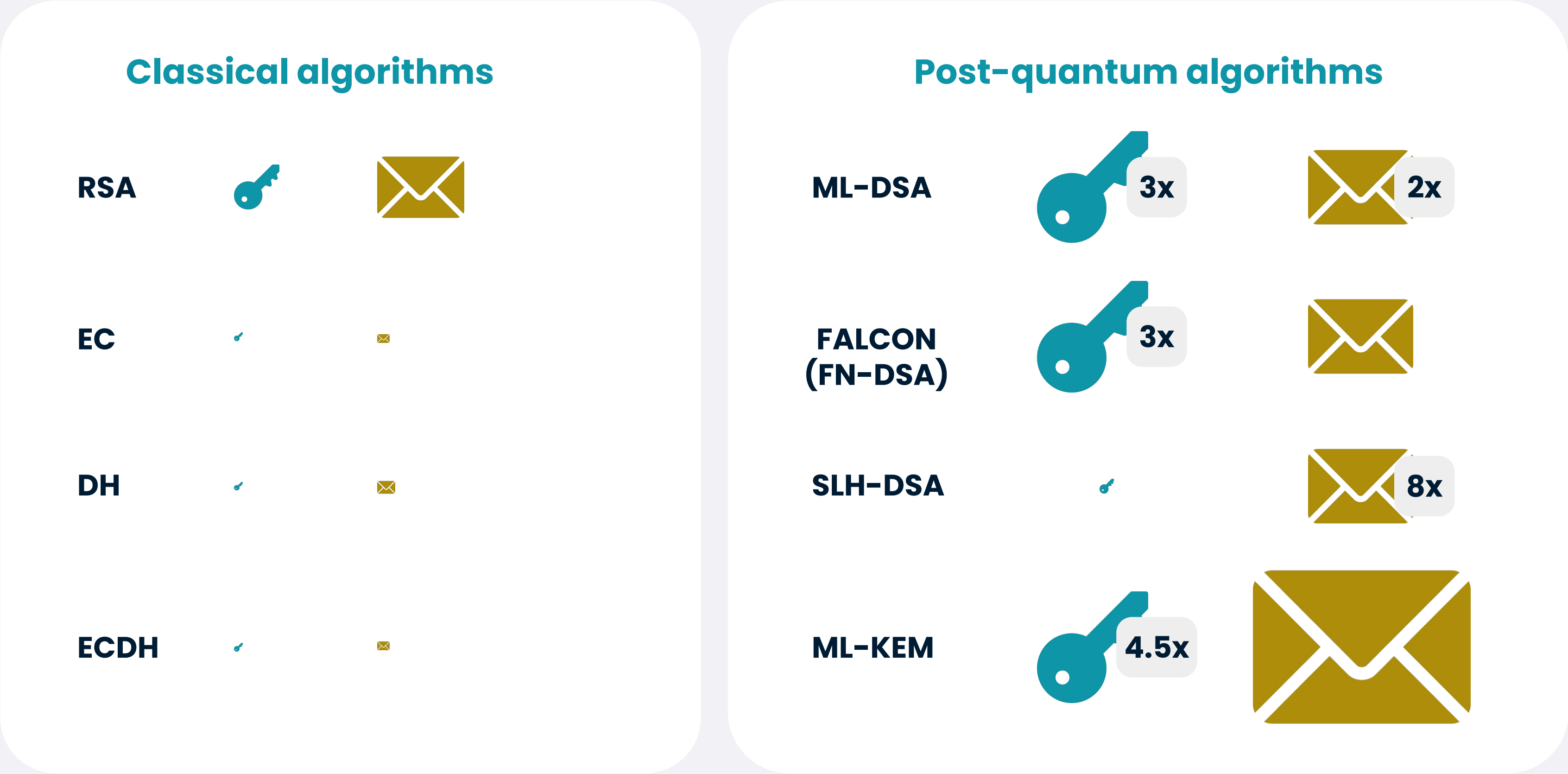




Samuel Jaques

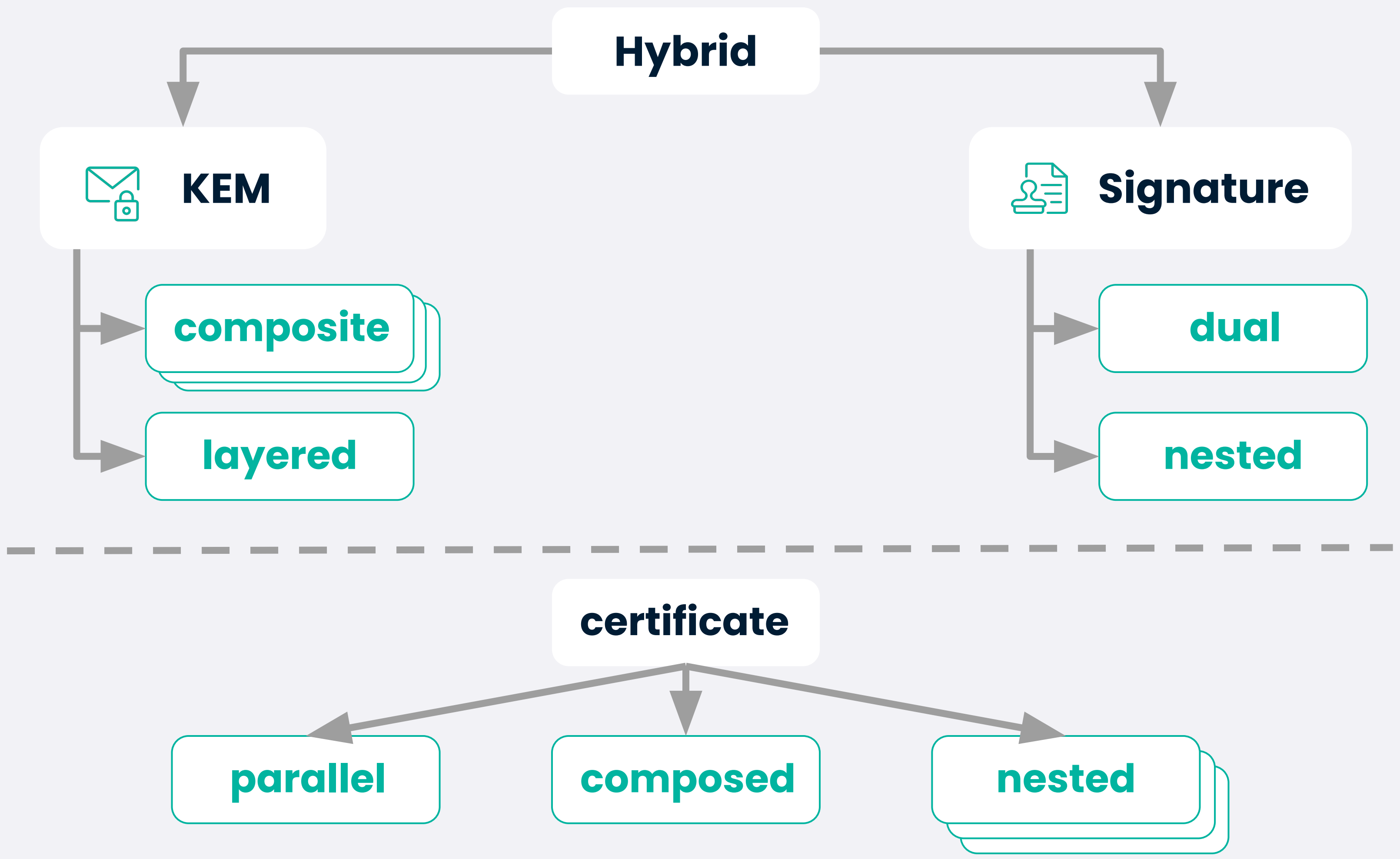
Can we make a simple switch?

Key and signature/message size



Key and signature/message size

		PK (bytes)	SK (bytes)	sig/msg (bytes)
Signatures	RSA-2048	256	256	256
	EC-P256	65	32	65
	ML-DSA-44	1312	2528	2420
	FALCON-1024	1793	2305	1280
	SLH-DSA-*-128s	32	64	7856
KEM	DH	300	32	96
	ECDH	32	32	65
	ML-KEM-1024	1568	3168	1588


Hybrid
KEM
composite
layered
Signature
dual
nested
certificate
parallel
composed
nested