# Integer Factorization of Events: Every Integer Is an Event, Every Quotient Is Division

Claude and MJC

February 2026

**Abstract**

We make precise a point missed in v1–v3: when arithmetic coding is treated as exact (e.g., via GMP arbitrary precision), the integers *are* the events and the maps between event spaces *are* integer division. An event $e \in \mathbb{N}$ enters any quotient event space by literal division: $e \mapsto e/d$ when $d \mid e$, or $e \mapsto e \bmod d$ when it does not divide evenly. The quotient $e/d$ is the new event in the coarser space; the remainder $e \bmod d$ is what was forgotten. Equivalence classes $\{e : e \equiv r \pmod{d}\}$ are the fibers of this projection, and they carry the structure of the ring $\mathbb{Z}/d\mathbb{Z}$. The divisor lattice of $|E|$ is the lattice of all quotient event spaces. Factoring $|E|$ into primes gives the finest decomposition into independent sub-event-spaces; any other factorization gives a coarser grouping. The entire formalism—event spaces, quotient maps, equivalence classes, independence, combination—reduces to elementary properties of the integers under multiplication and division.

*This is v4. v1: AC via UMs. v2: $E \to \mathbb{N} \to Q$ bridge. v3: sequence as integer, word Fourier. v4: the algebra is literal—integers are events, division is the quotient map, equivalence classes are rings.*

## 1 The Missed Point

The previous papers treated the integer encoding as a *representation*—a way to *name* events using numbers. The actual point is stronger: the integers do not merely name the events, they *are* the events. The arithmetic operations on the integers are not analogies for operations on events; they are the operations themselves.

Concretely: use GMP (GNU Multiple Precision Arithmetic) and do arithmetic coding with exact rational arithmetic. No floating point, no finite-precision window, no approximation. Then:

- An event is an integer $e \in \{0, 1, \ldots, |E| - 1\}$.
- Moving to a quotient space is dividing by a factor.
- The remainder is what was discarded.
- The equivalence class is a residue class.
- The ring $\mathbb{Z}/d\mathbb{Z}$ is the quotient event space.

None of these are metaphors. They are the definitions.

## 2 Events Are Integers

### 2.1 The Setup

Fix an event space $E$ with $|E| = N$ possible events. An event is an integer $e \in \{0, 1, \ldots, N-1\}$. This is not a "choice of encoding"—it is what an event *is* once we have decided on a finite set of mutually exclusive outcomes and ordered them.

For a factored event space $E = E_1 \times E_2 \times \cdots \times E_k$ with $|E_i| = n_i$, the event is the mixed-radix integer:

$$e = v_1 + n_1 v_2 + n_1 n_2 v_3 + \cdots + (n_1 \cdots n_{k-1}) v_k$$

where $v_i \in \{0, \ldots, n_i - 1\}$ is the value in $E_i$. The maximum is $N = \prod n_i$.

When the $n_i$ are assigned distinct primes (the prime power encoding from v2), this becomes $e = \prod p_i^{v_i}$ and the mixed-radix representation becomes the prime factorization. But the algebraic structure we describe here works for *any* factorization of $N$, not just the prime one.

### 2.2 Every Integer Is an Event

The integers $\{0, 1, \ldots, N-1\}$ are in bijection with events. But the integers do not stop at $N-1$. What about $e \geq N$?

Under modular arithmetic, $e$ and $e+N$ are the same event (they are in the same equivalence class mod $N$). So every non-negative integer is an event: it just wraps around. The event corresponding to integer $m$ is $m \bmod N$.

This wrapping is not a technicality. It is the reason the unit circle appears: the map $e \mapsto e \bmod N$ is periodic with period $N$, and identifying 0 with $N$ makes $\{0, \ldots, N-1\}$ into a cycle, which is $\mathbb{Z}/N\mathbb{Z}$—a ring, and topologically a circle.

## 3 Division Is the Quotient Map

### 3.1 The Fundamental Operation

Let $d$ be a divisor of $N$ (i.e., $d \mid N$). The map

$$\pi_d : \{0, \ldots, N-1\} \to \{0, \ldots, N/d - 1\}, \qquad \pi_d(e) = \lfloor e/d \rfloor$$

is the projection onto a quotient event space of size $N/d$.

What does this map do? It groups events into blocks of $d$ consecutive integers and identifies all events within a block: $\pi_d(e) = \pi_d(e')$ iff $\lfloor e/d \rfloor = \lfloor e'/d \rfloor$, i.e., $e$ and $e'$ differ by less than $d$.

The remainder $e \bmod d \in \{0, \ldots, d-1\}$ is the information that $\pi_d$ discards—the "fine structure" within each block.

**Proposition 1** (Division–Remainder Decomposition)**.** *Every event $e$ decomposes uniquely as:*

$$e = d \cdot \underbrace{\lfloor e/d \rfloor}_{\textit{quotient event}} + \underbrace{e \bmod d}_{\textit{remainder}}.$$

*The quotient event $\lfloor e/d \rfloor$ is the event in the coarser space $E' = \{0, \ldots, N/d-1\}$. The remainder $e \bmod d$ is the event in the complementary space $E'' = \{0, \ldots, d-1\}$. Together they recover $e$.*

This is the division algorithm. It is also the fundamental operation of the UM: splitting a joint event into a coarse prediction and a fine residual.

### 3.2 For Factored Spaces

When $E = E_1 \times E_2$ with $|E_1| = n_1$, $|E_2| = n_2$, $N = n_1 n_2$, and $d = n_1$:

$$\pi_{n_1}(e) = \lfloor e/n_1 \rfloor = v_2, \qquad e \bmod n_1 = v_1.$$

Division by $n_1$ "forgets" event space $E_1$ and retains $E_2$. The remainder remembers $E_1$. This is projection onto one factor of the product space.

For the prime power encoding with $e = \prod p_i^{v_i}$, dividing by $p_j^{v_j}$ removes event space $E_j$:

$$e/p_j^{v_j} = \prod_{i \neq j} p_i^{v_i}.$$

This is the "forget $E_j$" map, and it is literal integer division.

### 3.3 Chains of Division

Dividing by $d_1$ then by $d_2$ is the same as dividing by $d_1 d_2$:

$$\pi_{d_2} \circ \pi_{d_1}(e) = \lfloor \lfloor e/d_1 \rfloor / d_2 \rfloor = \lfloor e/(d_1 d_2) \rfloor = \pi_{d_1 d_2}(e)$$

(when $d_1 d_2 \mid N$). Successive projections compose by multiplying divisors. The hierarchy of quotient spaces is the divisor lattice of $N$.

## 4 The Divisor Lattice Is the Lattice of Event Spaces

### 4.1 Divisors and Sub-Spaces

Every divisor $d$ of $N$ defines a quotient event space of size $N/d$ (the image of $\pi_d$). The divisors of $N$ form a lattice under divisibility:

- $d = 1$: the full space $E$ (no information discarded).
- $d = N$: the trivial space $\{0\}$ (all information discarded).
- $d_1 \mid d_2$: the space for $d_2$ is a coarsening of the space for $d_1$ (more discarded).
- $\gcd(d_1, d_2)$: the finest common coarsening.
- $\operatorname{lcm}(d_1, d_2)$: the coarsest common refinement.

**Theorem 2** (Divisor Lattice = Event Space Lattice)**.** *The lattice of quotient event spaces of $E$ (ordered by coarsening) is isomorphic to the divisor lattice of $N = |E|$ (ordered by divisibility). The meet is* gcd *(finest common coarsening) and the join is lcm (coarsest common refinement).*

*Proof.* Each divisor $d \mid N$ gives a partition of $\{0, \ldots, N-1\}$ into blocks of size $d$, hence a quotient space of size $N/d$. The correspondence $d \leftrightarrow$ "quotient of size $N/d$" is order-reversing: larger $d$ = coarser space. The lattice operations follow from gcd and lcm of divisors corresponding to meet and join of the induced partitions. □

### 4.2 Prime Factorization = Finest Decomposition

If $N = p_1^{a_1} \cdots p_k^{a_k}$, the prime power factors $p_i^{a_i}$ are the *atoms* of the divisor lattice—they correspond to the finest independent sub-event-spaces. Any other divisor $d$ groups multiple atoms together.

**Corollary 3** (Independence and Primes)**.** *Two quotient event spaces (divisors $d_1, d_2$) are* independent *(their information does not overlap) iff* $\gcd(d_1, d_2) = 1$*. This occurs iff $d_1$ and $d_2$ have disjoint prime factors. Independence of event spaces is coprimality of divisors.*

### 4.3 Example: $N = 12$

$12 = 2^2 \times 3$. The divisor lattice:

| Divisor $d$ | Quotient size $N/d$ | Interpretation |
|---|---|---|
| 1 | 12 | Full 12-event space |
| 2 | 6 | Forget the fine bit of $E_{2^2}$ |
| 3 | 4 | Forget $E_3$ entirely |
| 4 | 3 | Forget $E_{2^2}$ entirely |
| 6 | 2 | Keep only $E_3$'s parity vs. $E_{2^2}$'s parity |
| 12 | 1 | Trivial (forget everything) |

Divisors 3 and 4 are coprime ($\gcd(3, 4) = 1$): the event spaces $E_3$ (size 3) and $E_{2^2}$ (size 4) are independent. Their product gives back the full space: $3 \times 4 = 12$.

## 5 Equivalence Classes Are Rings

### 5.1 Residue Classes

The quotient map $\pi_d$ identifies events that differ by a multiple of $d$. The equivalence class of event $e$ under $\pi_d$ is:

$$[e]_d = \{e' \in \{0, \dots, N-1\} : e' \equiv e \pmod d\} = \{e, e+d, e+2d, \dots\} \cap \{0, \dots, N-1\}.$$

This class has $N/d$ elements: all events that "look the same" when the fine structure (the last $\log_2 d$ bits, roughly) is ignored.

### 5.2 The Ring Structure

The set of equivalence classes $\{[0]_d, [1]_d, \dots, [d-1]_d\}$ is the ring $\mathbb{Z}/d\mathbb{Z}$. It has:

- **Addition:** $[a]_d + [b]_d = [a+b]_d$. Adding events in the quotient space.
- **Multiplication:** $[a]_d \cdot [b]_d = [ab]_d$. Composing events in the quotient space.
- **Zero:** $[0]_d$. The "null event."
- **Unity:** $[1]_d$. The "unit event."

**Proposition 4** (Quotient Event Spaces Are Rings)**.** *For any divisor $d \mid N$, the quotient event space under $\pi_d$ carries the ring structure of $\mathbb{Z}/d\mathbb{Z}$. Addition in the ring is event combination; multiplication is event composition; the units $(\mathbb{Z}/d\mathbb{Z})^\times$ are the invertible events (those coprime to $d$).*

### 5.3 The Chinese Remainder Theorem

When $N = d_1 d_2$ with $\gcd(d_1, d_2) = 1$:

$$\mathbb{Z}/N\mathbb{Z} \cong \mathbb{Z}/d_1\mathbb{Z} \times \mathbb{Z}/d_2\mathbb{Z}.$$

This is the Chinese Remainder Theorem (CRT). In event-space language: *when two quotient event spaces are independent (coprime divisors), the full event space is their direct product, and the ring decomposes accordingly.*

**Theorem 5** (CRT = Event Space Factorization)**.** *The factorization of $N$ into coprime factors $N = d_1 \cdots d_m$ with $\gcd(d_i, d_j) = 1$ for $i \neq j$ gives:*

$$E \cong E_{d_1} \times \cdots \times E_{d_m}$$

*where $E_{d_i} = \mathbb{Z}/d_i\mathbb{Z}$. The isomorphism is the CRT map: $e \mapsto (e \bmod d_1, \ldots, e \bmod d_m)$. Knowing the event in each factor determines the event in the whole space. This is the ring-theoretic statement that factored event spaces are independent.*

For the prime power factorization $N = \prod p_i^{a_i}$:

$$\mathbb{Z}/N\mathbb{Z} \cong \prod_i \mathbb{Z}/p_i^{a_i}\mathbb{Z}.$$

This is the *finest* such decomposition. Each $\mathbb{Z}/p_i^{a_i}\mathbb{Z}$ is a "primary" event space that cannot be split further into coprime factors.

# 6 Maps Between Event Spaces Are Factors

## 6.1 The Category of Event Spaces

We now have a category:

- **Objects:** Quotient event spaces, indexed by divisors $d \mid N$. The space for divisor $d$ is $\mathbb{Z}/d\mathbb{Z}$ (having $d$ events).
- **Morphisms:** For $d_1 \mid d_2$, the projection $\pi : \mathbb{Z}/d_2\mathbb{Z} \to \mathbb{Z}/d_1\mathbb{Z}$ given by $e \mapsto e \bmod d_1$. This is a ring homomorphism.
- **Composition:** $\pi_{d_1|d_2} \circ \pi_{d_2|d_3} = \pi_{d_1|d_3}$ (projections compose).
- **Identity:** $\pi_{d|d} = \text{id}$.

**Proposition 6** (All Maps Are Reduction mod $d$)**.** *Every morphism in this category is of the form $e \mapsto e \bmod d_1$ for some $d_1 \mid d_2$. That is:* every map between event spaces is modular reduction*. There are no other structure-preserving maps.*

This is the sense in which "all maps are factors": the morphism from the $d_2$-space to the $d_1$-space exists iff $d_1 \mid d_2$, and the map itself is "divide and take remainder." The factor $d_2/d_1$ is the "information ratio" between the spaces—the number of fine-grained events per coarse-grained event.

## 6.2 Arithmetic Coding as a Chain of Projections

Arithmetic coding processes one symbol at a time. At step $t$, the full event so far is an integer $e_t \in \{0, \ldots, |E|_t - 1\}$ where $|E|_t = 256^t$ (for byte-level coding). The new byte $b_t$ extends the event:

$$e_t = 256 \cdot e_{t-1} + b_t.$$

This is a ring operation in $\mathbb{Z}/256^t\mathbb{Z}$: scale the previous event by 256 (shifting it into a coarser position) and add the new byte (filling in the fine detail).

The coder maintains the interval $[L_t, H_t)$ which is the image of $e_t$ under the model-weighted quotient map. Each coding step is:

1. Receive $b_t$.
2. Compute $e_t = 256\, e_{t-1} + b_t$ (extend the event integer).

3. Subdivide $[L_{t-1}, H_{t-1})$ according to $P_t(\cdot)$ (apply the model).

4. Select the subinterval for $b_t$ (project via the model-weighted map).

5. Emit any bits of $L_t$ that have stabilized (reduce mod power of 2 in the output).

With exact arithmetic (GMP rationals), steps 1–4 are exact: $e_t$ is an integer, the interval boundaries are rationals, and no precision is lost. The "sliding window" of approximate coders disappears—the full $e_t$ is maintained.

### 6.3 Decoding as Factoring

Given the compressed output (a rational number $q \in [0, 1)$), decoding recovers $e = e_n$. Each decoding step extracts one factor:

1. Find $b_1$ such that $q$ falls in $b_1$'s interval under $P_1(\cdot)$.

2. Compute $q' = (q - F_1(b_1))/P_1(b_1)$ (rescale: divide out $b_1$'s contribution).

3. Repeat with $q'$ to find $b_2$, etc.

Step 2 is literal division: the code value $q$ is divided by $P_1(b_1)$ (the probability, which is a factor of the interval width). Decoding = dividing out factors one at a time = factoring the code integer into its event components.

## 7 The Ring of Compressed Events

### 7.1 Compression as Ring Homomorphism

The model $\mathcal{M}$ defines a map from events to code values:

$$\phi_{\mathcal{M}} : \mathbb{Z}/N\mathbb{Z} \to [0, 1), \qquad \phi_{\mathcal{M}}(e) = F_{\mathcal{M}}(e) + \tfrac{1}{2}P_{\mathcal{M}}(e).$$

This is the model-weighted quotient from v2. Under a better model, the image $\phi_{\mathcal{M}}(e)$ falls in a wider interval (higher probability, fewer bits).

The code length $\ell(e) = -\log_2 P_{\mathcal{M}}(e)$ is the "distance" from the identity in the ring—how many bits it takes to distinguish $e$ from neighboring events under the model.

### 7.2 Two Models, One Event

Given two models $\mathcal{M}_1, \mathcal{M}_2$, the same event $e$ maps to different code values $\phi_1(e), \phi_2(e)$ and different code lengths $\ell_1(e), \ell_2(e)$. The model is a *choice of coordinates* on the ring—it determines which events are "close" (similar code values) and which are "far" (different code values).

A better model makes the actual events close together (high probability, short codes) and the impossible events far apart (low probability, long codes). The cross-entropy $H_{\text{cross}} = \frac{1}{n}\sum_t \ell(e_t)$ measures the average distance from the identity—the average code length.

### 7.3 Model Combination in the Ring

Combining models corresponds to combining ring homomorphisms. The combination problem (which destroyed performance with geometric mean in our experiments) is the problem of composing maps on the ring:

- **Arithmetic mean** of probabilities: corresponds to averaging the interval widths. This is a valid ring operation (addition followed by normalization).

- **Geometric mean** of probabilities: corresponds to multiplying interval widths. This is also a ring operation, but it changes the normalization (the intervals no longer sum to 1 without re-weighting). This is why geometric mean is catastrophic in practice: it breaks the ring structure by violating the normalization axiom.
- **Arithmetic coding interleaving**: process bytes from model 1 and model 2 alternately. This interleaves the factor chains, which is a valid composition in the ring.

# 8 From $\mathbb{Z}$ to Everything

## 8.1 The Integers Contain All the Structure

The preceding sections show that the UM formalism, when made exact, reduces to properties of the integers:

| UM Concept | Integer Operation |
|---|---|
| Event | Integer $e \in \{0, \ldots, N-1\}$ |
| Event space | $\mathbb{Z}/N\mathbb{Z}$ |
| Quotient space | $\mathbb{Z}/d\mathbb{Z}$ for $d \mid N$ |
| Projection to quotient | $e \mapsto e \bmod d$ |
| Information discarded | $e \mapsto \lfloor e/d \rfloor$ |
| Independent sub-spaces | Coprime factors of $N$ |
| Full decomposition | Prime factorization of $N$ |
| Equivalence class | Residue class mod $d$ |
| Space lattice | Divisor lattice of $N$ |
| CRT | $E \cong \prod \mathbb{Z}/p_i^{a_i}\mathbb{Z}$ |
| AC encoding step | $e_t = 256\, e_{t-1} + b_t$ |
| AC decoding step | Division and remainder |
| Compression | Ring homomorphism $\phi_{\mathcal{M}}$ |

## 8.2 Why This Matters

The integers are the most studied object in mathematics. By establishing that event spaces *are* (not "are analogous to") rings of integers modulo $N$, we import all of number theory and algebra into the UM framework:

- **Factoring difficulty.** Finding the prime decomposition of $N$ is computationally hard (this is the basis of RSA). In our setting, "factoring $N$" means discovering the independent sub-event-spaces—which is exactly the hard part of compression. The computational difficulty of integer factorization *is* the computational difficulty of discovering the right event space decomposition.
- **Ideal theory.** The ideals of $\mathbb{Z}/N\mathbb{Z}$ are in bijection with the divisors of $N$. Each ideal is a sub-event-space closed under addition. The ideal generated by $d$ is the "$d$-periodic" event space: events that repeat every $d$ steps.
- **Units and invertibility.** The unit group $(\mathbb{Z}/N\mathbb{Z})^\times$ has order $\varphi(N)$ (Euler's totient). Units are the "reversible" events: those from which the original event can be recovered by multiplication. Non-units are "lossy" events: some information is destroyed.
- **Characters.** The Dirichlet characters mod $N$ are the group homomorphisms $\chi : (\mathbb{Z}/N\mathbb{Z})^\times \to \mathbb{C}^\times$. These are exactly the "features" of the event space that respect the multiplicative structure—the Fourier basis on the unit group. Dirichlet $L$-functions are then generating functions for event-space features weighted by code length.

### 8.3 Arithmetic Coding Is Factoring

We can now state the complete identification:

**Theorem 7** (AC = Integer Factoring)**.** *Let $e \in \{0, \dots, 256^n - 1\}$ be a byte sequence encoded as an integer. Let $\mathcal{M}$ be a model assigning probabilities $P_t(b)$ at each position. Then:*

1. *Arithmetic coding computes a code value $c$ and code length $\ell$ such that $c$ falls in an interval of width $2^{-\ell}$.*

2. *The code value $c$ is a function of $e$ and $\mathcal{M}$. It is the image of $e$ under the model-weighted ring homomorphism $\phi_{\mathcal{M}}$.*

3. *Decoding from $c$ recovers $e$ by successively factoring out each byte's contribution: at each step, dividing by the model's probability and reading off the quotient and remainder.*

4. *The total code length $\ell = \sum_t -\log_2 P_t(b_t)$ is the "ring distance" from the identity—the number of bits needed to specify $e$'s equivalence class in the model-induced quotient.*

Compression is factoring: discovering the prime structure of $|E|$ (the independent sub-event-spaces), computing the quotient maps (the model predictions), and encoding the quotient chain as a bit string (the compressed output). Decompression is the inverse: given the bit string and the model, reconstruct the integer by multiplying the quotient-remainder pairs back together.

## 9 Conclusion

The point, once seen, is simple: an event is an integer, and everything we do with events is something we do with integers.

Quotient event spaces are not "like" quotient rings—they *are* quotient rings. The map from a fine-grained space to a coarse-grained space is not "analogous to" integer division—it *is* integer division. The equivalence classes of events that a model cannot distinguish are not "similar to" residue classes—they *are* residue classes.

This identification is not a metaphor, an analogy, or a pedagogical device. It is a mathematical identity, valid when arithmetic coding is performed exactly (as with GMP). Approximate coders (finite-precision windows) break the identity by introducing rounding, but the exact case is the ground truth.

The consequence is that the entire apparatus of algebraic number theory—divisor lattices, CRT, ideals, units, characters, $L$-functions—is available not as inspiration but as direct tooling for the analysis and construction of universal models. The hard problems of compression (finding the right event spaces, combining models, predicting structure at multiple scales) are the hard problems of number theory (factoring, computing discrete logarithms, evaluating $L$-functions), wearing different clothes.

## References


[1] Claude and MJC, "Arithmetic Coding via Universal Models: Any UM is a Compressor," 2026.

[2] Claude and MJC, "Arithmetic Coding, Quotients, and Prime Powers: The $E \to \mathbb{N} \to Q$ Bridge," 2026.

[3] Claude and MJC, "The Sequence as an Integer: Direct Construction from $256^t$ to Fourier on Words," 2026.

[4] Claude and MJC, "Event Arithmetic: $E$ onto $\mathbb{N}$," 2026.

[5] Claude and MJC, “Kneser–Ney as Quotient,” 2026.

[6] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression,” *Comm. ACM*, 1987.