

# Polylog6: Hierarchical Equilateral Polyform Compression and Discovery

Polylog6 redefines geometric data management, treating **equilateral polyform compression** as a hierarchical discovery process governed by strict geometric rules. Our node-based architecture enables precise assembly and exploration of diverse polyforms, from fundamental Johnson and Archimedean solids to complex compound structures.

## The Four Pillars of Polylog6 Geometry:



### Uniform Edge Lengths

All polygons maintain **identical unit edge lengths**, ensuring foundational consistency across all constructions like Johnson solids.



### Exact Edge Connections

Connections are strictly **vertex-to-vertex, edge-to-edge**, preserving inherent symmetry in forms like the Cuboctahedron.



### Scaling by Edge Count

Larger forms are created by **more unit-length edges**, not longer ones, allowing scalable combination of polyforms.

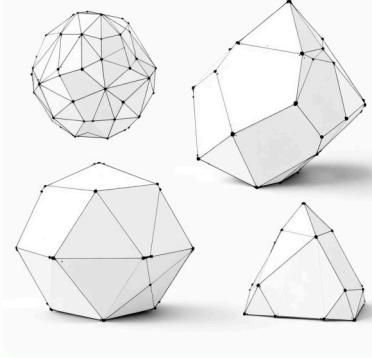


### Undeformed Folding

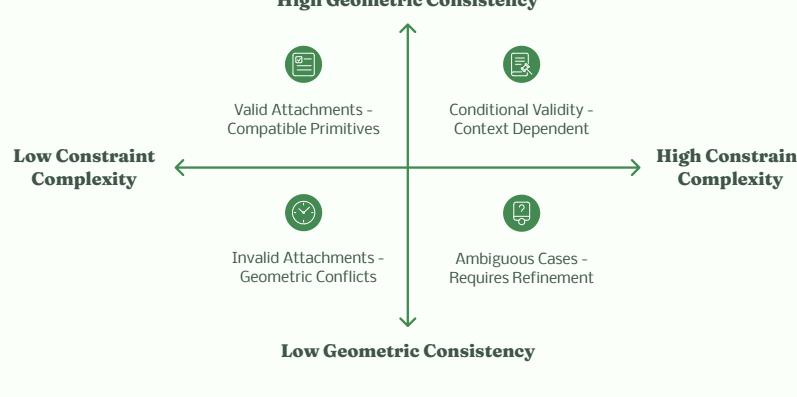
Folding occurs **only at shared edges**, with no internal deformation, critical for geometric integrity, seen in the Snub Cube.

## Geometric Principles in Action:

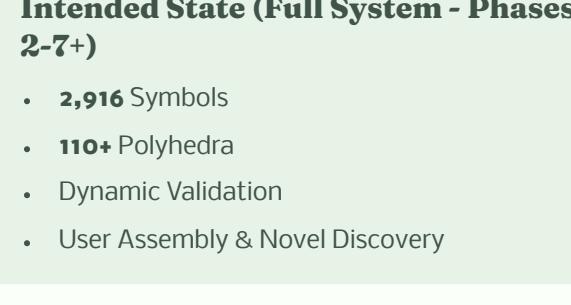
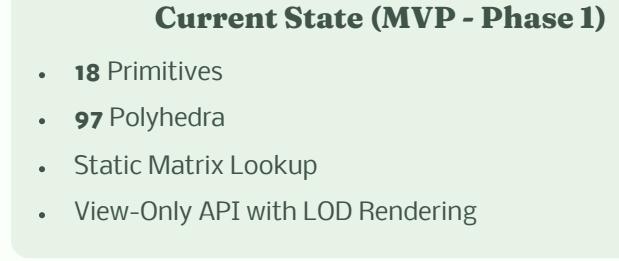
Below are visual examples of how Polylog6 applies these principles to construct and categorize polyforms. Our system also inherently maintains the geometric relationships and symmetry that define these structures, facilitating both accurate digital representations and the discovery of new, geometrically sound forms.



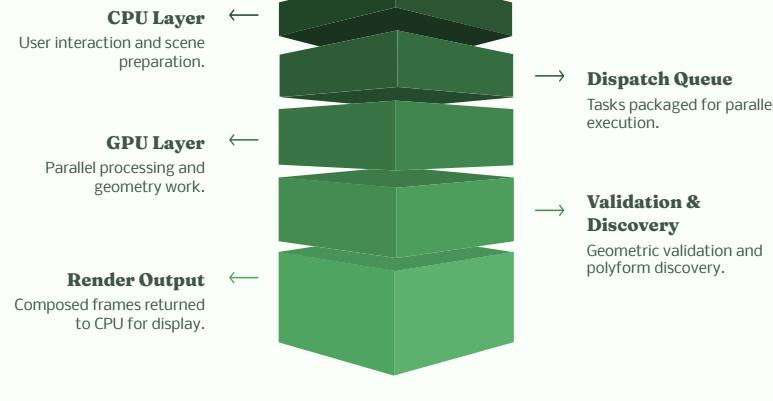
The Polylog6 system relies on a sophisticated 18x18 Attachment Matrix to define all possible valid connections between its primitive polyforms. This matrix acts as the foundational rule-set for ensuring geometric integrity during polyform assembly and discovery.



## System Evolution: From MVP to Full Discovery



The architecture supporting this evolution spans both local processing for user interaction and powerful server-side computation for complex validation and discovery.



This innovative approach moves beyond simple compression to a comprehensive system for building and exploring complex equilateral polyform compositions with unparalleled efficiency and unwavering geometric accuracy.

[Explore the Technology](#)

[View Research Paper](#)

- For a comprehensive transformation including the detailed technical slides (Edge Matching Matrix, Edge Matching Flow, Face Topology, Closure Detection, System Architecture, Implementation Roadmap), please submit follow-up requests for each specific slide you wish to create. This current card provides a high-level overview and clarifies the core geometric principles and system states.

# Polylog6: Core Insight - Equilateral Polyform Reducibility

Our system's foundation is built on the inherent reducibility of complex polyforms into standardized, perfectly equilateral polygon nodes. This enables both precise construction and novel discovery.



## Rigid Equilateral Principles

- Built from **perfectly equilateral polygons** (triangles, squares, hexagons, etc.).
- Identical unit edge length** for all edges in the workspace.
- Precise **edge-to-edge attachment** ensures compatibility.
- Undeformed folding** only at shared edges; individual polygons remain pristine.
- Naturally preserves **geometric symmetry** of forms like Johnson and Archimedean solids.



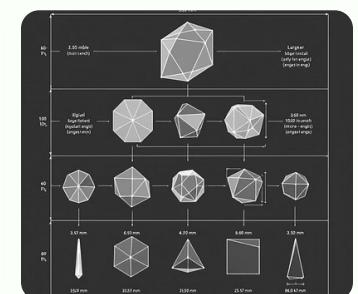
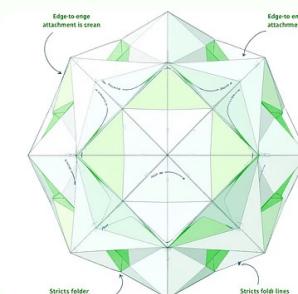
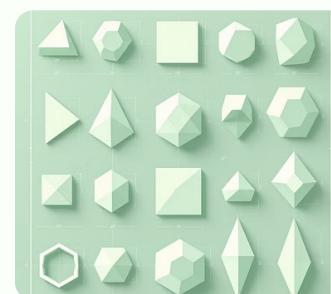
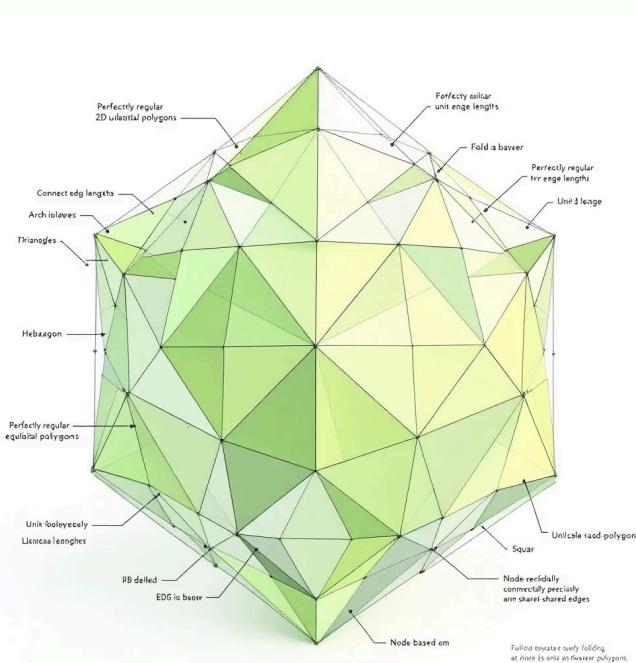
## Node-Based Discovery System

- Each equilateral polygon acts as a **standardized node**.
- Enables **efficient hierarchical encoding** and discovery of complex polyforms.
- Scalable building block approach:** Larger polygons are physically larger due to more unit-length edges, not longer edges.
- Supports **novel compound polyforms** and scaled assemblies (e.g., combining tetrahedrons).



## Validated & Applied Representation

- GPU/CPU architecture ensures **strict geometric validation** and rendering.
- Transforms polyforms into **efficient, Unicode-based representations** for tiered classification.
- Guarantees adherence to **geometric regularity**, non-deformation, and inherent symmetry.
- Precisely models forms like the **cuboctahedron** and **snub cube**, maintaining integrity through unit edge and angular rules.



The gallery illustrates basic equilateral polygons and their hierarchical assembly into complex forms. Strict edge-to-edge attachment and folding rules are maintained, ensuring perfect geometric regularity and symmetry. Larger polygons are larger due to a greater number of identical unit edges, facilitating scalable **compound polyforms**.

Complex polyhedral structures, like **Johnson** or **Archimedean solids**, are formed by perfectly equilateral polygons with identical unit edge lengths, connected strictly edge-to-edge without deformation. This node-based system allows efficient encoding and maintains inherent symmetry.

## Polylog6: Tier 0 Primitives – MVP (18) vs. Vision (2,916+)

The foundational "Tier 0" of Polylog6 defines the universal vocabulary of primitive **equilateral polyforms**. Each is rigorously defined for geometric consistency and efficient encoding, forming the bedrock for both our current MVP and future expansions.

### Strict Geometric Consistency

All edges are of identical unit length across the entire workspace (e.g., triangle, square, hexagon edges are precisely the same length). Each polygon maintains its perfect geometric shape with equal internal angles.

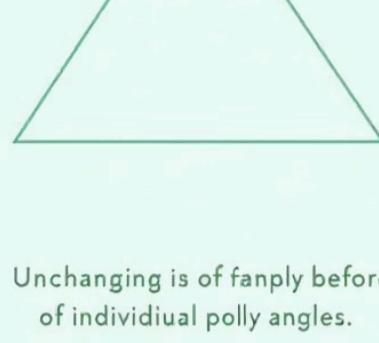
### Strict Edge-to-Edge Attachment

Polygons connect seamlessly without deformation or overlap, enabling robust hierarchical encoding and symbolic representation.

## EQUILATERAL POLYGONS

### TRIANGLE

All sides are equal unit length and perfect internal angles.



Unchanging shape before individual polygon angles.

### SQUARE

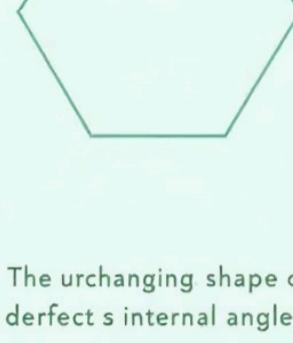
All sides are equal unit length and perfect internal angles.



Use unchanging shape and after attachment.

### HEXAGON

All sides are equal unit length and perfect internal angles.



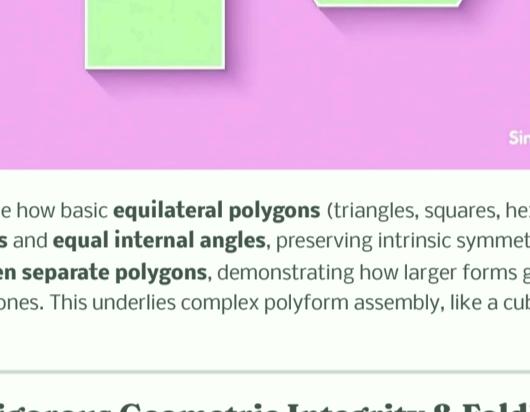
The unchanging shape of perfect internal angles.

 plippe  
constervicys.com

This standardization is critical for the Phase 1 MVP, ensuring accurate construction and representation of complex structures like **Johnson solids** (e.g., square pyramids) and **Archimedean solids** (e.g., cuboctahedron).

### Fun is assembly diagram

Your triangle, square to the equal sides have can, edge-to-equivalent sides is connect edge- and assembly ready assembly.



SindLine

Visualize how basic **equilateral polygons** (triangles, squares, hexagons) are built with **identical unit edge lengths** and **equal internal angles**, preserving intrinsic symmetry. They connect strictly edge-to-edge between **separate polygons**, demonstrating how larger forms grow by having more unit-length edges, not longer ones. This underlies complex polyform assembly, like a cuboctahedron.

### Rigorous Geometric Integrity & Folding Rules

- Identical Unit Length for All Edges:** Consistently applied across the entire workspace.
- Equal Internal Angles:** Ensures **perfect geometric consistency** of each polygon.
- Strict Edge-to-Edge Attachment:** From vertex to vertex **between separate equilateral polygons**, preventing deformation or arbitrary bending.

This allows complex forms like **Archimedean solids** while guaranteeing each **equilateral polygon remains a perfect, undeformed shape**.

### Core Principle: Node-Based Equilateral System

#### Square

Square: identical unit edge length and equal right angles; joins precisely edge-to-edge to triangles and hexagons, larger than a triangle due to more unit edges, not longer edges.

#### Triangle

Equilateral triangle: identical unit edge length and equal internal angles; connects edge-to-edge without deformation to form composite shapes.

#### Hexagon

Regular hexagon: identical unit edge length and equal internal angles; connects seamlessly edge-to-edge to build larger lattices-larger than square by edge count, not edge length.

Polylog6 prevents arbitrary shape manipulation or deformation. Folding occurs **ONLY** at shared edges **between distinct polygons**, maintaining perfect geometric integrity within each individual polygon. This is critical for structures like cuboctahedra and snub cubes.

### Unicode Encoding Readiness (Phase 1 Foundation)

- Identical unit edge length** and **equal internal angles** facilitate discrete Unicode symbols (e.g.,  $\Omega$ ,  $\Delta$ ) for each primitive **equilateral polyform**.
- Compressed Representation:** Efficient storage and transmission of 3D data.
- O(1) Validation:** Folding exclusively at shared edges between distinct polygons ensures rapid validation.

This capability is crucial for encoding structures from simple **compound polyforms** to complex **Johnson solids**, preserving their inherent symmetry.



## Tier 0 Structure: 18 Primitives (MVP) vs. 2,916+ Fundamental Symbols (Vision)

### Current MVP (Phase 1)

#### 18 Primitives

- 3 Base Polygons:** Triangle, Square, Hexagon
  - 15 Derived Polyforms:** Stable combinations up to 3 connected polygons.
- Focuses on essential building blocks for static matrix lookup and view-only functionalities.

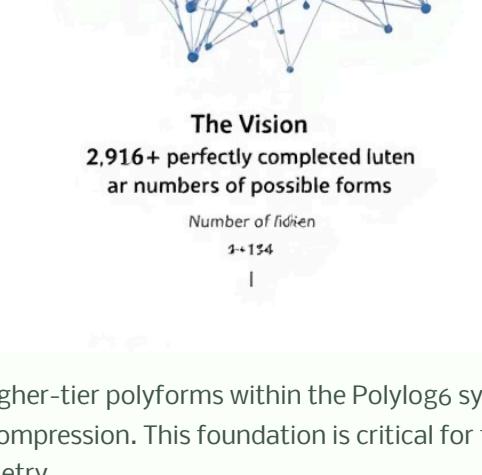
### Intended Vision (Phases 2-7+)

#### 2,916+ Symbols

- 36 Single Polygons:** All regular, convex polygons up to N edges.
- 288 Equilateral Polyform Pairs:** Stable 2-polygon combinations.
- 1,944 Equilateral Polyform Chains:** Stable 3-polygon sequences.
- 648+ Advanced Combinations:** Higher-order and specialized forms.

Enables dynamic validation, user assembly, and extensive polyform discovery.

### 2,916+ Perfectly connected equilateral Possible forms



This robust vocabulary forms the backbone for constructing all higher-tier polyforms within the Polylog6 system. It emphasizes geometric precision, **equilateral polygon**-based assembly where all edges are identical unit length, and symbolic compression. This foundation is critical for the current Phase 1 (MVP) and lays the groundwork for future dynamic assembly capabilities, ensuring the preservation of intrinsic symmetry.

# Polylog6: Tier 0 Primitive Vocabulary (Current MVP vs. Intended Vision)

The foundational "Tier 0" of Polylog6 establishes a universal geometric vocabulary for building complex structures. It's built on three core, visually enforced principles, consistent across both current MVP and future vision:



## Strictly Equilateral Polygons

All edges, from a triangle to a hexagon, share an identical unit length across the entire workspace.



## Undeformable Nodes

Each polygon maintains its perfect geometric integrity; larger polygons are simply composed of more identical unit-length edges.



## Precise Edge-to-Edge Attachments

Folding occurs only at shared edges between distinct polygons, enabling seamless and non-deforming assembly.

This rigorous architecture ensures robust hierarchical encoding and geometric validation by our specialized GPU/CPU pipeline. It guarantees the preservation of inherent symmetry and relationships found in known polyforms like **Johnson Solids** (e.g., square pyramids) and **Archimedean Solids** (e.g., cuboctahedron), critical for predictable construction and stable identification.



## Current MVP: 18 Primitives

The initial MVP focuses on 18 fundamental single equilateral polygons, sufficient for core functionalities like static matrix lookups and basic polyhedra construction. This limited set provides a stable foundation for Phase 1.



## Intended Vision: 2,916 Symbols

The full vision expands to 2,916 unique Unicode symbols, encompassing single, paired, and chained equilateral polygons, enabling dynamic validation and comprehensive user assembly capabilities beyond the MVP.



## Encoding & Validation

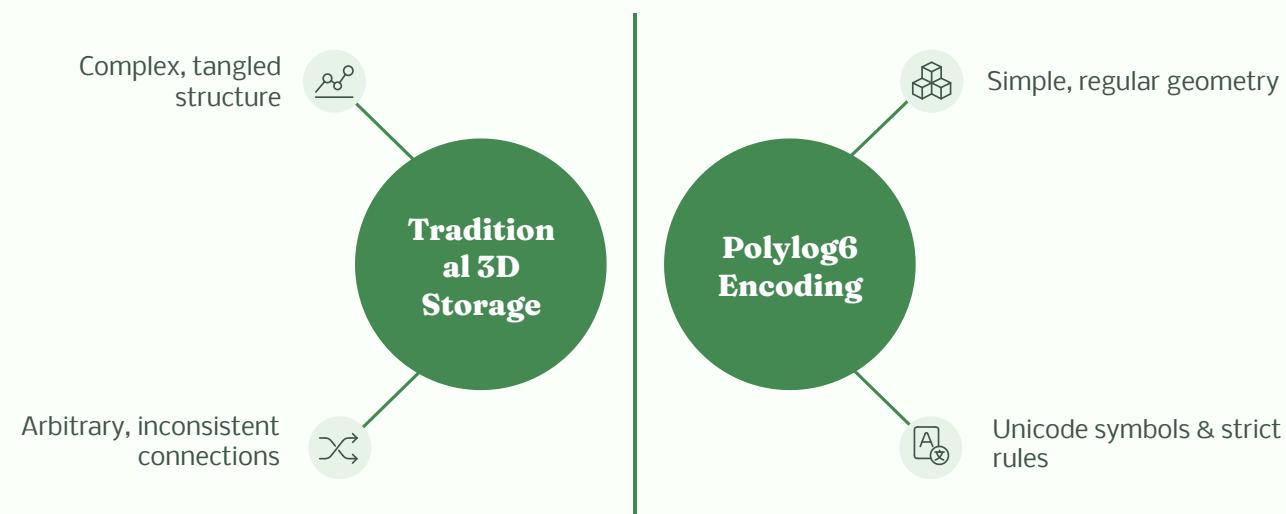
This comprehensive vocabulary (from the 18 MVP primitives to the 2,916 vision symbols) supports efficient, deterministic encoding and geometric validation, foundational for higher-tier polyform construction.

The expanded Tier 0 Primitive Vocabulary (Intended Vision) includes **2,916** unique Unicode symbols ( $\Omega_0$ ), allowing for the efficient and deterministic encoding of perfectly equilateral geometric primitives. This robust foundation ensures geometric validation and rendering by a specialized GPU/CPU architecture, laying the bedrock for all higher-tier polyform construction while inherently preserving their symmetry factors.



# Polylog6: The Geometric Encoding Revolution

Polylog6 introduces a groundbreaking system for encoding **perfectly regular, equilateral polyforms** where **all edges across the entire workspace are of identical unit length**. This enables unprecedented precision, compression, and discovery of novel geometric structures, ranging from fundamental **Johnson solids** to complex **Archimedean solids**.



## The Problem: Traditional 3D Storage

### Arbitrary Connections

Compromises geometric integrity with internal deformations.

### Inconsistent Generation

Lacks rule-based methods for novel geometric structures.

### Inefficient Representation

Struggles with highly symmetrical forms without strict rules.

- ❑ **Traditional storage:** A simple 3D cube often requires **~3 KB** for vertex data, mesh topology, and materials. Inefficient for complex structures without strict geometric rules.

## The Solution: Polylog6 Encoding

### Strict Edge-to-Edge

Connections maintain identical unit edge length consistency.

### Symmetry Preservation

Constraints maintain inherent symmetries (cuboctahedron, snub cube).

### Undeformable Nodes

Folding only at shared edges; polygons retain rigid shapes.

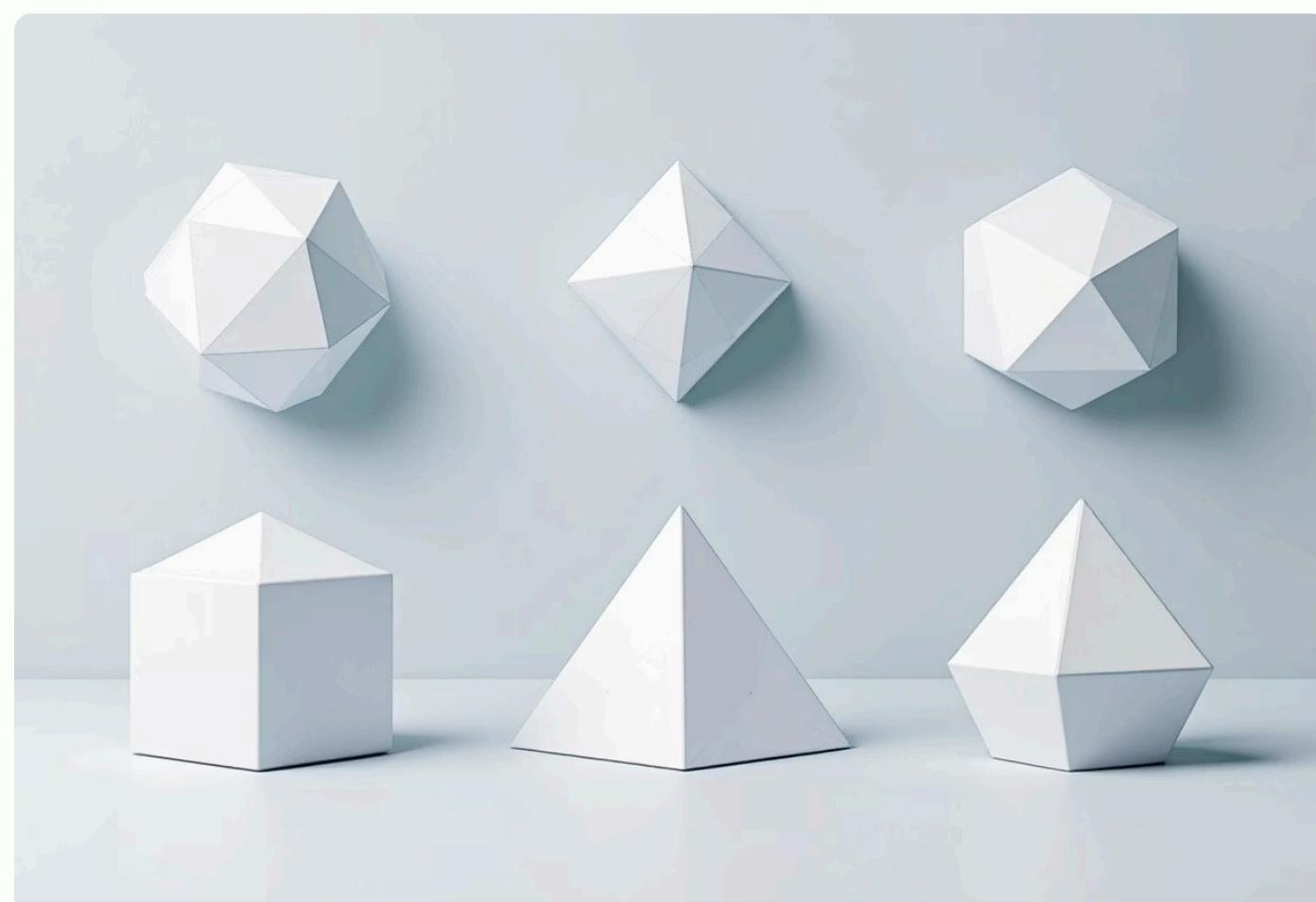
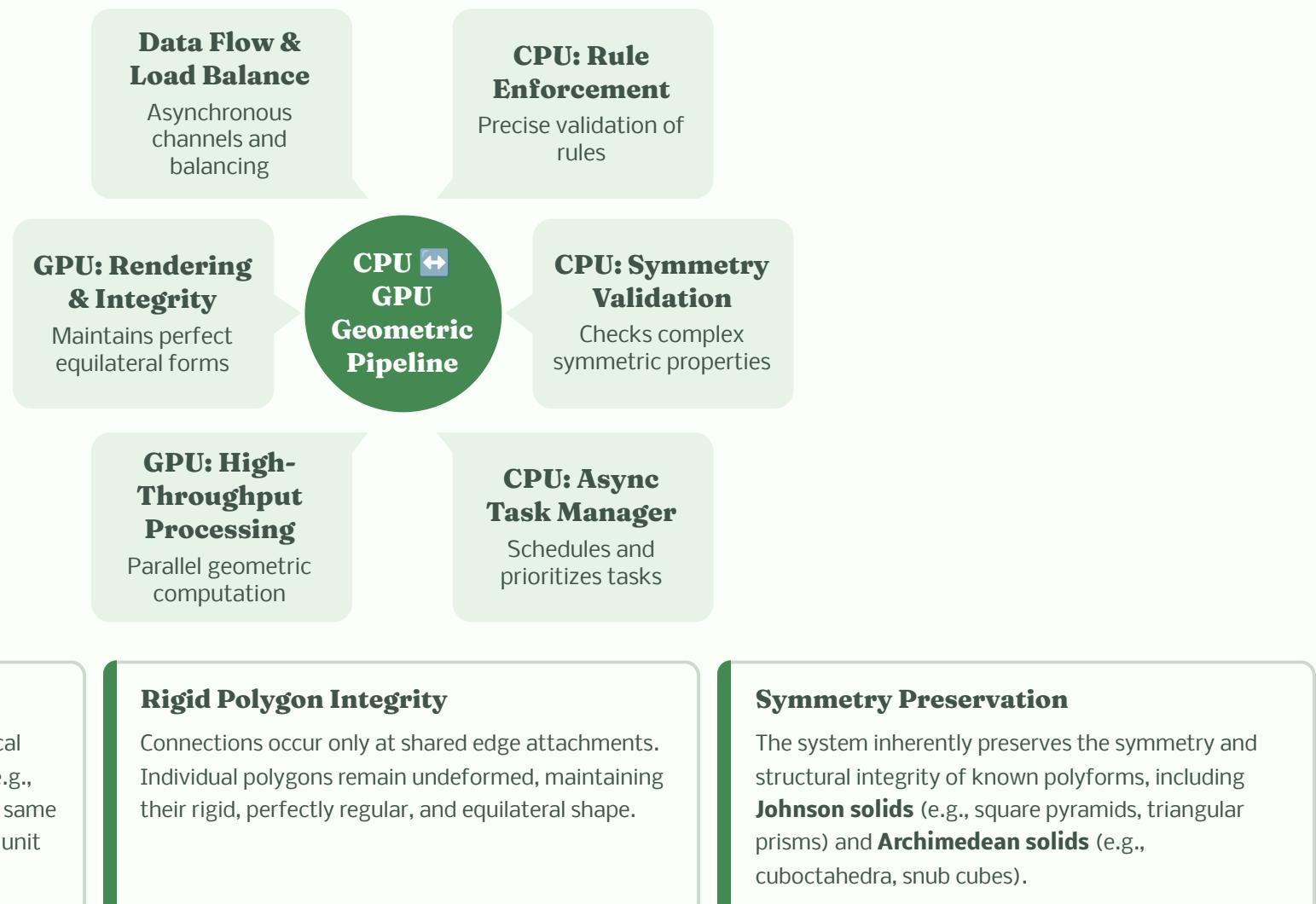
### Scalable Assemblies

Precise construction of compound and scaled polyforms.

- ❑ **Polylog6 Advantage:** The same 3D cube, encoded with Unicode symbols, can be represented in as little as **~2 bytes**, achieving unprecedented compression.

# Polylog6: GPU/CPU Asynchronous Architecture for Geometric Integrity & Discovery

The Polylog6 system leverages a sophisticated asynchronous architecture to distribute computational load between CPU and GPU, ensuring optimal performance for geometric integrity and discovery. This layered approach minimizes latency and maximizes throughput while guaranteeing all polyforms adhere to critical geometric rules.



This architecture underpins both our MVP (Phase 1) and the advanced capabilities of the Full System (Phases 2-7+), enabling the discovery and precise encoding of novel, highly symmetric geometric structures.

Key performance metrics highlight the system's efficiency in maintaining geometric accuracy for **perfectly equilateral polygons with all edges identical unit length**, particularly for complex polyform assemblies and their inherent symmetry. This applies to both MVP (Phase 1) and our future phases of custom polyform assembly and discovery:

**500K+**

**Polyforms/sec**

Sustained processing throughput for complex **equilateral polyform operations** (e.g., Johnson and Archimedean solids), maintaining perfect geometric integrity, unit edge length, and **inherent symmetries**.

**2ms**

**Avg. Latency**

Minimal average latency for critical polyform validation and rendering tasks, ensuring no deformation and perfect edge-to-edge fits, even for intricate structures like the cuboctahedron.

**1.2GB**

**VRAM Usage**

Optimized memory footprint for large-scale, rule-abiding equilateral geometric data on GPU, **efficiently handling the complexity of compound and scaled polyforms**.

**15%**

**Avg. CPU Util.**

Efficient GPU offloading keeps CPU utilization low, allowing the CPU to focus on **precise equilateral polygon rule enforcement and validating complex symmetric properties**.

## System State Comparison: MVP (Phase 1) vs. Full System (Phases 2-7+)

### Current State (MVP - Phase 1)

- Static edge matching via  $18 \times 18$  matrix for **individual Johnson and Archimedean solids**.
- Fixed face topology per polyhedron (no merging).
- $O(1)$  validation using pre-computed stability scores.
- View-only capability (no user assembly creation).
- 97 polyhedra accessible via API** with LOD rendering, each maintaining precise equilateral and symmetric properties.
- Supports **18 primitives**.

### Intended State (Full System - Phases 2-7+)

- Real-time edge matching validation during user assembly, enabling **complex composite forms** (e.g., a rectangular hexagon from squares on a hexagon).
- Dynamic face contact detection and merging (e.g., 2 triangles  $\rightarrow$  1 quad), **critical for scalable compound polyforms**.
- Closure detection ( $o$  boundary edges = closed assembly), **validating integrity of complete symmetric structures**.
- Runtime symbol generation and tier promotion.
- Custom fold angles and **novel polyform discovery**, exploring new symmetric arrangements.
- Aims for **110+ polyhedra**.
- Supports **2,916 symbols**.

# Polylog6: Tier 0 - Equilateral Primitives & Unicode Index

The foundation of Polylog6 rests on its [Tier 0](#) primitive vocabulary, precisely indexed via a FastAPI-powered Unicode system. This tier defines the elementary **equilateral geometric building blocks**, crucial for constructing symmetrical polyforms like Johnson and Archimedean solids. Key principles:

## Equilateral Polygons

Strict edge-to-edge attachment with equal internal angles.

## Consistent Unit Edge Length

All edges are identical unit edge length across the entire workspace, ensuring seamless compatibility.

## Perfect Shapes

Polygons maintain perfect equilateral shapes; no deformation or stretching is permitted.

## Folding Rules

Precise folding applies **only at shared edges between distinct polygons**, not within individual ones.

This immutable base, characterized by perfect geometric regularity, is essential for the hierarchical construction and symmetry preservation of all higher-order polyforms.

## Tier 0 Primitive Vocabulary: Building Blocks



### Single Polygons: 36

The core **18 elementary equilateral polygons** (e.g., triangle, square, hexagon) and their mirrored forms, each with **identical unit edge length**.



### Polygon Pairs: 324

Combinations of two **equilateral polygons strictly connected edge-to-edge**, maintaining **perfect equilaterality** and **zero deformation**.



### Polygon Chains: 2,556

Sequences of three **equilateral polygons connected at shared edges**, encoding rules for 3D projection and compound polyforms.

**Total Tier 0 Equilateral Primitives: 2,916**

These primitives are encoded into unique Unicode symbols for compact representation and efficient property validation. This hierarchical encoding enables rapid discovery and validation of new polyform structures while ensuring strict geometric integrity and **unwavering equilaterality and consistent unit edge lengths**.

- ❑ **Tier 0** forms the immutable base layer, ensuring geometric integrity and consistency across all higher tiers of polyform discovery within the Polylog6 system through its strict adherence to:
  - **Equilateral polygons:** With equal internal angles.
  - **Edge-to-edge connections:** Always preserving unit edge length.
  - **Non-deforming shapes:** Maintaining perfect equilaterality.
  - **Symmetry preservation:** Allowing for construction of known polyforms like the Cuboctahedron, where polygon size is determined solely by the number of unit edges.

# Polylog6: Regular Polyforms & Immutable Geometric Architecture

Polylog6 exclusively uses **regular polygons** as its foundational, immutable geometric building blocks. These polygons adhere to strict criteria ensuring perfect regularity and non-deforming shapes:

## Consistent Unit Edge Length

All internal edges within each polygon maintain an **identical unit length** across the entire workspace (e.g., triangle, square, hexagon edges are precisely the same length).

## Perfect Regularity & Angles

Polygons maintain their precise, regular shape (**no stretching, bending, or distortion**) with equal internal angles (e.g., 60° for a triangle, 90° for a square).

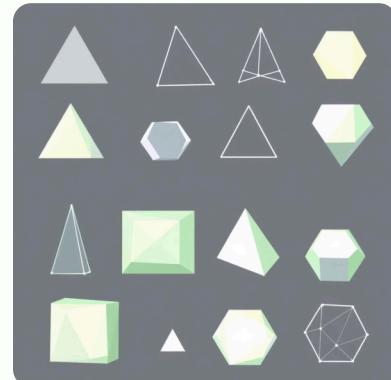
## Edge-to-Edge Attachment

Polygons attach **edge-to-edge** at their boundaries. Folding occurs solely along these shared edges between separate polygons, never within a polygon itself.

## Fundamental Polyform Series & Hierarchical Structure

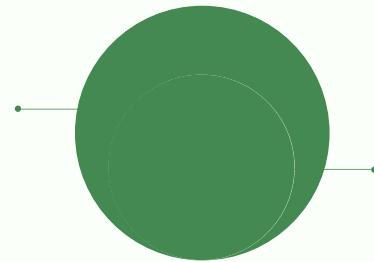
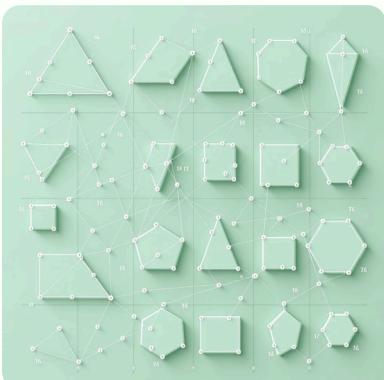
### Building Blocks: Series A-D

- Series A & B:** Define 18 **regular polygons** (3 to 20 unit-length edges) as core Tier 0 primitives.
- Series C & D:** Provide redundant sets of 18 **regular polygons**, enabling diverse assembly pathways.
- Unit-Edge Counts:** Dictate immutable attachment points and geometrically regular bonding between **non-deforming polygons**.

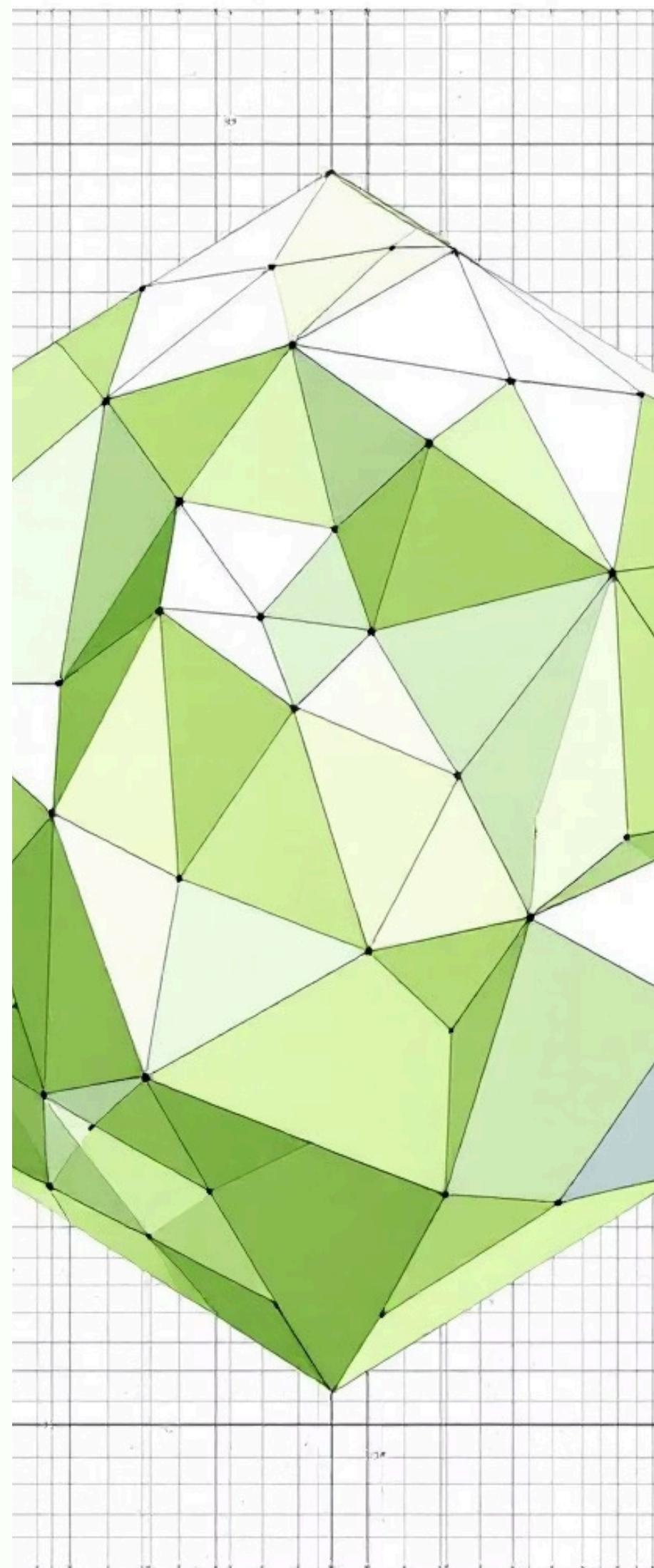


### Unicode Encoding & Immutable Matrix

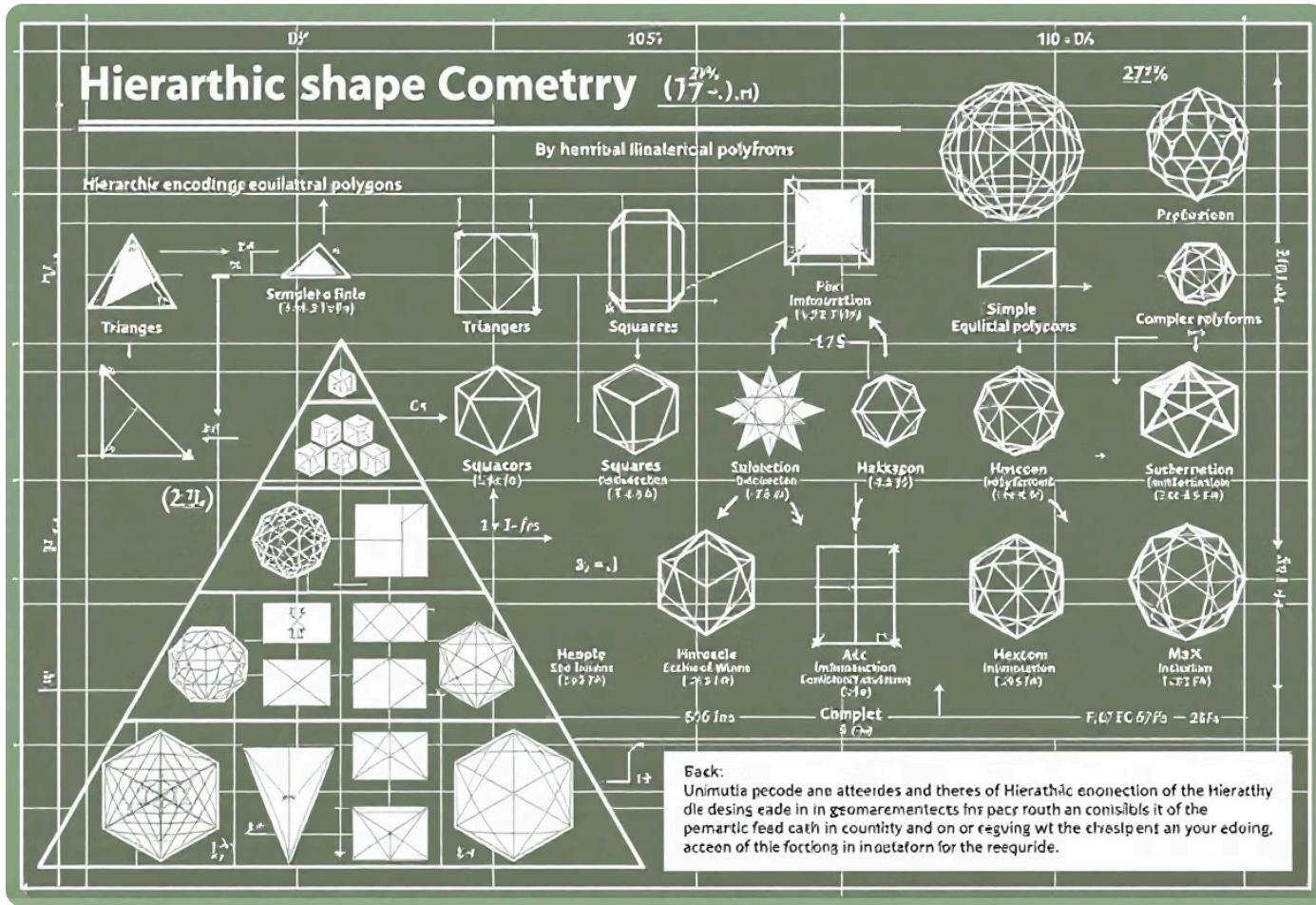
Strategic design of Series A, B, C, and D is crucial for the Unicode-based **regular polygon** encoding system. This dual coverage approach ensures robust validation and construction of complex **regular polyforms** with **zero deformation** and consistent **unit edge lengths**.



This tiered system progresses from **perfectly regular** primitives (Tier 0) to known polyhedra (Tier 1), candidate structures (Tier 2), and promoted forms (Tier 3). It ensures systematic discovery and validation of **regular polyforms** under strict geometric rules and consistent **unit edge lengths**. Each **regular polygon's** Unicode representation defines its inherent immutable properties and potential **edge-to-edge** attachment points, forming an **immutable attachment matrix** for all Polylog6 assembly.



# Polylog6: Hierarchical Regular Polyform Encoding



The Polylog6 system uses **Unicode encoding** to represent and reconstruct **perfectly regular polyform assemblies**. This ensures immutable geometric integrity with **unit-length edges** and **equal internal angles** at every level, preventing deformation.

## Tier 0: Primitive Polygons

Base Unicode characters for fundamental regular polygons (e.g., triangle, square, hexagon).

**1**

**Current MVP:** 18 primitives

**Intended Vision:** 2,916 symbols (all possible planar regular polygons)

These form the immutable foundation with **unit-length edges** and **perfectly equal angles**.

## Tier 1: Known Polyhedra

Combinations of Tier 0 primitives form basic 3D regular polyhedra (e.g., tetrahedron, cube, octahedron).

**2**

**Current MVP:** 97 polyhedra accessible via API

**Intended Vision:** 110+ canonical polyhedra (Johnson, Archimedean, Platonic solids)

Validated against known forms to preserve **geometric regularity** and **edge-to-edge attachment**.

## Tier 2: Candidate Polyforms

Recursive assembly of Tier 1 polyhedra or other candidates, forming novel **geometrically regular structures**. This tier focuses on generating and evaluating new potential polyforms for stability and geometric integrity.

**3**

**Current MVP:** Limited, experimental generation

**Intended Vision:** Automated generation and validation pipeline

## Tier 3: Promoted Polyforms

Validated geometric discoveries promoted for further exploration and integration. These advanced polyforms adhere to all strict **regular polyform** constraints, expanding the Polylog6 design space.

**4**

**Current MVP:** Manual promotion of verified forms

**Intended Vision:** Dynamic promotion with unique Unicode symbols

Explore the visual foundation of Polylog6's geometric encoding:

# Polylog6: Encoding the Universe

## Foundational Geometric Principles and Construction Rules

Polylog6 leverages a revolutionary Unicode-based encoding system for geometric structures. It ensures faithful digital representation and discovery of complex polyforms by adhering to critical, physically accurate geometric principles:

### Equilateral Polygons as Nodes

Fundamental building blocks: equilateral triangles, squares, regular hexagons. Each acts as a perfect node.

### Identical Unit Edge Length

All edges across the entire workspace maintain an identical unit length, ensuring consistent scaling.

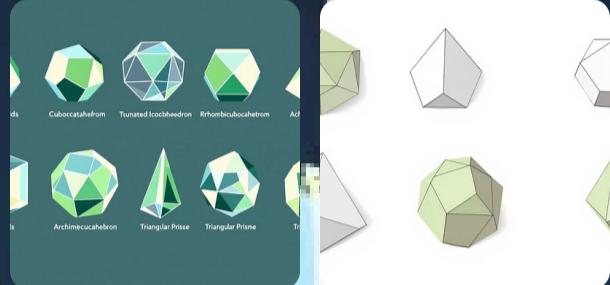
### Precise Edge-to-Edge Attachment

Polygons connect seamlessly along shared vertices and unit-length edges.

### No Deformation

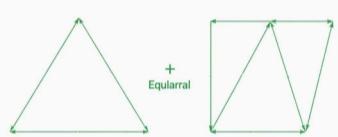
Polygons always maintain their perfect geometric regularity, **no irregular shapes are permitted**.

This rigorous adherence enables the precise representation of known forms like **Johnson Solids** and **Archimedean Solids**, and facilitates the discovery of novel **compound polyforms** all while preserving inherent symmetry and geometric integrity.



1

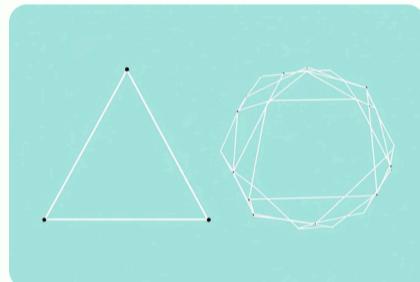
### Edge Connections



Attachments occur strictly edge-to-edge between **equilateral polygons** with **identical unit-length edges**. This forms structures like the **cuboctahedron** without deformation.

2

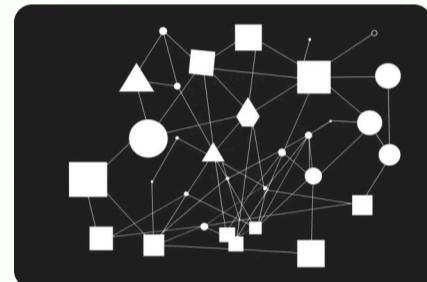
### Unit Edge Length



**ALL edges maintain identical unit length.** Larger polygons are proportionally larger due to more unit-length edges, not longer individual edges.

3

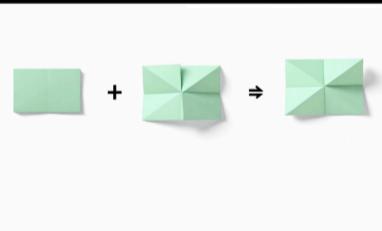
### Node Geometry



Each **equilateral polygon** is an independent node with perfect internal geometry, maintaining integrity through **identical unit-length edge** connections.

4

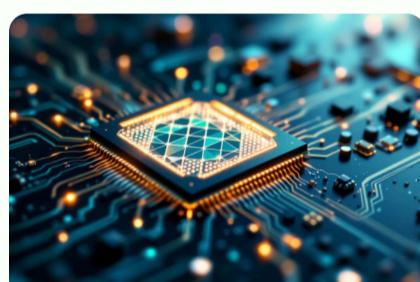
### Strict Folding Rules



Equilateral polygons fold only along **shared unit-length edges** to form 3D structures. **No stretching or bending** is permitted, ensuring structural fidelity.

5

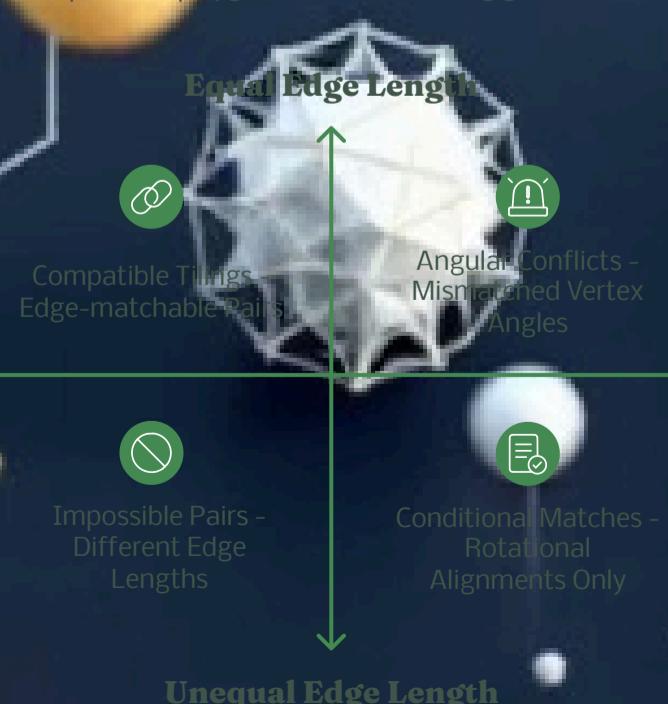
### GPU/CPU Validation



Robust architecture ensures real-time validation and rendering of all polyform structures, adhering to every geometric rule for **equilateral polygons** and **identical unit-length edges**.

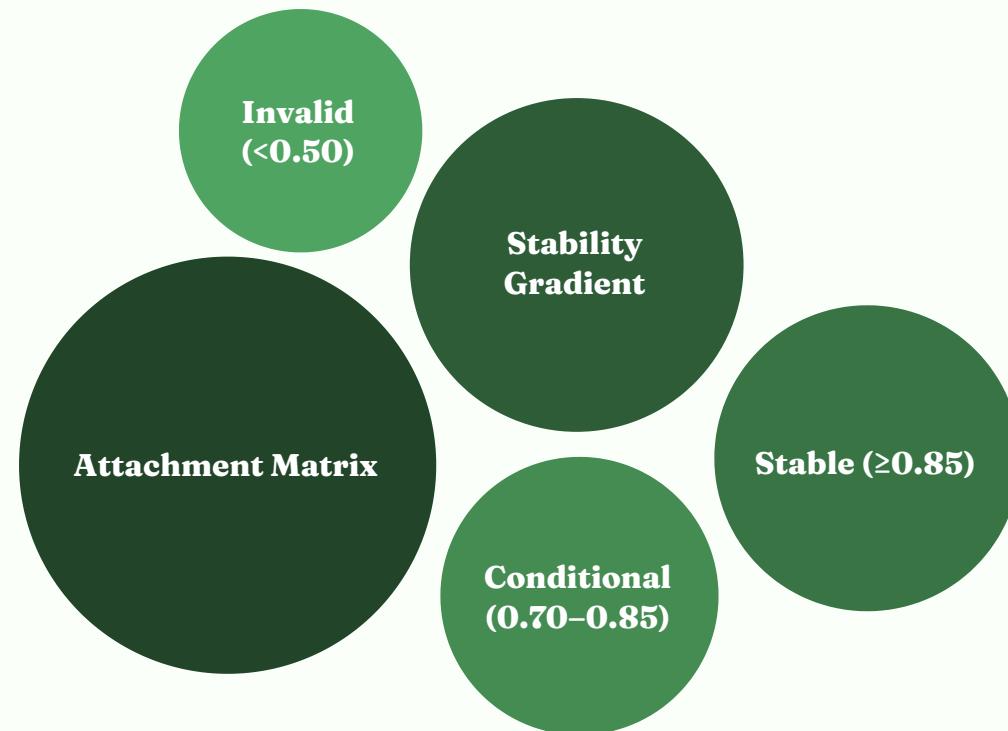
## The Polylog6 Attachment Matrix

The system relies on a foundational attachment matrix to define how primitive polygons connect, ensuring geometric integrity and predictability in polyform construction.



# Polylog6: 18x18 Attachment Matrix - Edge Connection Rules

The Polylog6 system employs a comprehensive 18x18 attachment matrix to govern how different polygon types connect, ensuring rigorous geometric integrity and predictability in polyform construction. This heatmap visualizes the stability and validity of all possible edge connections between pairs of polygons, based on precise angular alignment and unit edge length rules.



□ **Statistics:** 324 total pairs | 140 stable (43%) | 120 conditional (37%) | 64 invalid (20%)

**a<sub>3</sub> ↔ a<sub>3</sub>: 70.53°, stability 0.95**

Tetrahedral edge, perfect closure

**a<sub>3</sub> ↔ b<sub>2</sub>: 60°, stability 0.78**

Mixed attachment, context-dependent

**b<sub>2</sub> ↔ d<sub>6</sub>: stability <0.50**

Incompatible connection

**Note:** Data source: catalogs/attachments/attachment\_matrix.json

**Performance:** O(1) lookup, <100ns validation

# Polylog6: Edge Matching - Current MVP vs. Intended System

## Current MVP - Phase 1: Static Matrix Lookup

### Method:

- Pre-computed  $18 \times 18$  attachment matrix
- $O(1)$  lookup in JSON file
- 324 attachment options pre-calculated

### Validation:

- Stability threshold check ( $\geq 0.70$  or  $\geq 0.85$ )
- Simple numeric comparison
- No geometric computation
- Instant validation (<10ms)

### Limitations:

- User cannot specify custom fold angles
- Cannot validate arbitrary assemblies
- No edge-to-edge contact detection
- Static rules only

**Use Case:** View-only mode with predefined polyhedra

## Intended System - Phases 2-7+: Dynamic Geometric Validation

### Method:

- $O(1)$  lookup for pre-computed attachments
- $O(n)$  geometric computation for custom angles
- User specifies fold angle (not just Netlib)
- System validates closure

### Validation:

- Query: Will this fold angle close?
- Compute: Dihedral angle from 3D geometry
- Validate: Non-self-intersection check
- Cache: Store successful angles for reuse
- Real-time: <5ms per validation

### Capabilities:

- Custom fold angles (user-specified)
- Edge length scaling (polynomial growth)
- Multi-edge attachments (face merging)
- Closure detection (is assembly closed?)
- Novel discovery (detect new polyforms)

**Use Case:** User-driven assembly with intelligent validation

### Current MVP: Triangle-Square Attachment

Lookup shows "Valid at  $60^\circ$ " (from matrix)

### Intended System: Triangle-Square Attachment

User specifies  $72^\circ$ , system computes new dihedral angle, validates closure

# Polylog6: Face Matching - Topology Evolution & Closure Detection



## Static Face Topology

### Current State:

- Each polyhedron has fixed face list
- Faces defined by vertex sequences
- Face normals (all point outward)
- LOD simplification (faces removed, not merged)
- No face-to-face contact tracking

**Limitation:** Assembly = collection of separate objects, no merging



## Dynamic Face Contact Detection

### New Capability:

- Detect when 2 polyhedra have adjacent faces
- Check coplanarity (are faces in same plane?)
- Identify shared edges
- Calculate face normal alignment
- Determine if faces can merge

**Workflow:** User places 2 polyhedra near each other → System detects face proximity → Offers merge option → User confirms or rejects



Can these merge?



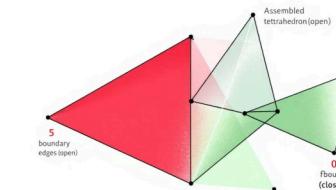
## Closed Assembly Detection

### Closure Analysis:

- Count boundary edges (unmatched edges)
- If count = 0: Assembly is CLOSED ✓
- If count > 0: Assembly is OPEN (can continue building)
- Validate: No holes, all normals outward

**Result:** CLOSED assembly = valid polyform

- Possible automatic promotion to Tier 2
- Trigger face merging optimization
- Calculate improved compression ratio



Edges highlighted RED (boundary) → GREEN (interior) as assembly closes.

"5 boundary edges (open)" → "0 boundary edges (closed)"

# Polylog6: Implementation Roadmap - 6 Phases to Full System



The roadmap outlines a horizontal progression through 6 phases. Phase 1 is complete (green), and phases 2-6 are pending (blue). We are currently at the transition point between Phase 1 and Phase 2.

Each phase transition is marked by specific "Go/No-Go" criteria, serving as success metrics for moving forward.

# Polylog6: Real-World Applications - Five Key Use Cases

## Education

- Students explore 97 known polyhedra
- Interactive 3D visualization
- Learn symmetry principles
- Export structures as Unicode

Example: "Build a cuboctahedron, see its 1,500:1 compression"



## Geometric Design

- Designers create tessellated facades
- Combine polyforms for structures
- Instant compression for collaboration
- Share via Unicode strings

Example: "Facade = octahedron + 4×tetrahedra, 1 Unicode symbol"



## Pattern Discovery

- Researchers identify novel polyforms
- Automatic tier promotion
- Publish using Unicode symbols
- Contribute to research library

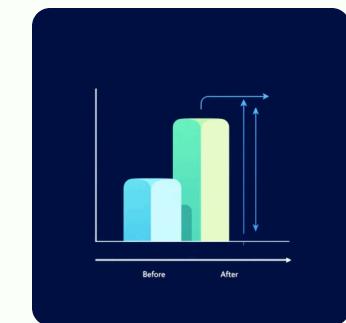
Example: "Novel compound discovered, published as  $\Omega_{125}$ "



## Compression & Storage

- Archive geometric libraries
- Efficient 3D model databases
- Real-time visualization streaming
- 1,500:1 compression ratio

Example: "1GB library  $\rightarrow$  700KB compressed"



## Collaborative Design

- Teams co-design polyforms
- Version history via git
- Instant reconstruction at checkpoints
- Share discoveries with one URL

Example: "Branch: feature/snub-cube-variant, diff: 2 bytes"

