

The Hunt for Legendre Pairs of Length 153

A structured GPU search for an open case of the Hadamard conjecture



Why this matters

1893

Hadamard's question: is there a ± 1 matrix with mutually orthogonal rows for every order divisible by 4? Still open.

668

the smallest order for which no Hadamard matrix is known

153

= 9×17 : the next open case of the prescribed-compression conjecture, now that 117 is settled

Legendre pairs are conjectured to exist for every odd length. If they do, the Hadamard conjecture follows. A pair of length 153 gives a structured Hadamard matrix of order 308.

Hadamard matrices at work

- Walsh–Hadamard codes separate the channels in CDMA mobile networks.
- Reed–Muller codes built from them carried Mariner 9's pictures of Mars.
- Hadamard masks multiplex light in spectrometers and single-pixel cameras.
- Plackett–Burman screening designs decide which factors matter in industrial experiments.
- Compressive sensing and quantum-information constructions lean on them too. The circulant kind a Legendre pair produces needs almost no storage and transforms through the FFT.

Whose research is this



Dr. Ilias S. Kotsireas

Professor, Physics and Computer Science
Wilfrid Laurier University, Waterloo, Ontario
Director, CARGO lab

web.wlu.ca/science/physcomp/ikotsireas

The mathematics behind this search comes from Dr. Kotsireas and his colleagues.

- Compression (Đoković and Kotsireas, 2015) turns a composite length into a known seed to uncompress. It is what makes lengths like 153 searchable at all.
- The prescribed-compression conjecture for lengths divisible by 9. Its 117 case was settled in 2026 (Blackthorn); $153 = 9 \times 17$ is next.
- Research in symbolic computation and combinatorial designs, using algorithmic and high-performance methods to settle existence questions.
- Editor-in-Chief, Journal of Algebraic Combinatorics; Fellow of the Institute of Combinatorics and its Applications.

Legendre pairs were introduced by Seberry and her students in 2001. Distributive builds and runs the distributed search.

Distributed search matters beyond 153

Quantum optimization benchmarks

Low-autocorrelation binary sequences (LABS). Proven optima stop at $N = 66$, set in 2016. JPMorgan, Argonne and Quantinuum used LABS for evidence of QAOA scaling advantage (Science Advances, 2024), and it is one of ten classes in the 2025 Quantum Optimization Benchmarking Library. Every new proven length is ground truth the field lacks.

Radar and sonar

Low autocorrelation means a sharp echo with weak side lobes, so the same sequences shape pulse compression. Golay complementary pairs, close cousins of Legendre pairs, also tame peak power in OFDM radios.

Software testing

Covering arrays: the smallest set of test cases that exercises every combination of a few settings. Smaller arrays mean fewer tests, and the best known tables improve through exhaustive, verifiable search.

Industrial experiments

Screening designs built from Hadamard matrices decide which factors matter in a manufacturing or lab experiment. Unusual run sizes still benefit from new constructions.

What carries over: exact slicing with a coverage ledger, results anyone can re-verify, GPU kernels that run in browsers and native workers, and relays that store sorted records and join them in one streaming merge.

Two sequences that complete each other



Periodic autocorrelation: slide a sequence against itself by s and count agreements minus disagreements.

	shift s					
	+1	+2	+3	+4	+5	+6
$C_A(s)$	+1	+1	-3	+1	-3	-3
$C_B(s)$	-3	-3	+1	-3	+1	+1
sum	-2	-2	-2	-2	-2	-2

Shown: a genuine pair of length 17, built from the Legendre symbol mod 17.

A Legendre pair of odd length ℓ :

$$C_A(s) + C_B(s) = -2 \quad \text{for every shift } s \neq 0$$

equivalently, in the frequency domain,

$$\text{PSD}_A(k) + \text{PSD}_B(k) = 2\ell + 2 \quad \text{for every frequency } k \neq 0$$

Why length 153

$\text{circ}(A)$	$\text{circ}(B)$
$\text{circ}(B)^T$	$-\text{circ}(A)^T$

two-circulant core, bordered: a Hadamard matrix of order $2\ell + 2$

308

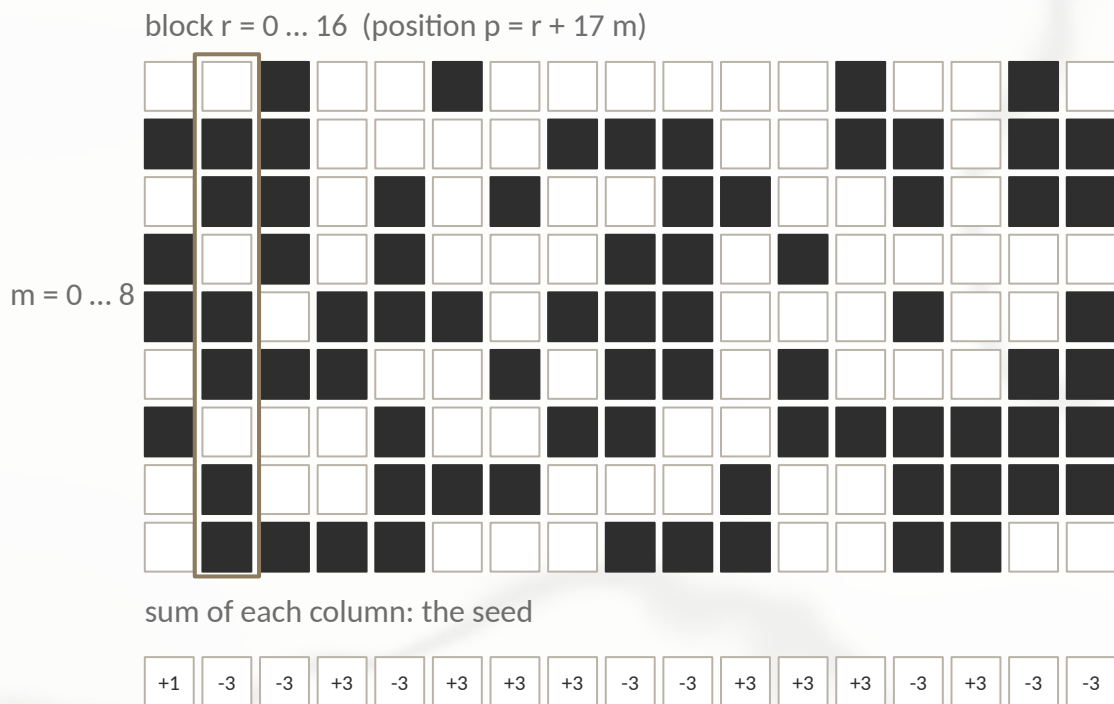
the Hadamard matrix a length-153 pair would produce $(2 \times 153 + 2)$

9×17

the next open case of the prescribed-compression conjecture for lengths divisible by 9

- Every odd prime length has a pair, straight from the Legendre symbol.
- For lengths $9p$, a conjecture of Kotsireas predicts pairs whose 9-compression is $\delta \pm 3\chi_p$, built from the quadratic character mod p .
- The case $p = 13$ (length 117) was settled in October 2026. The case $p = 17$ (length 153) is open.

Compression gives us a seed, and a space to search



9-compression: add the nine entries 17 apart. The conjecture fixes the result: $\delta - 3\chi_{17}$, a seed of length 17 with +1 first, then -3 on the squares mod 17 and +3 elsewhere.

Uncompressing is a choice per block:

- sum +3: six +1 and three -1 among nine positions, 84 ways
- sum -3: three +1 and six -1, 84 ways
- sum +1 (the first block): five +1 and four -1, 126 ways

$$126 \times 84^{16} \approx 7.7 \times 10^{32}$$

candidate sequences per side, before any constraint is applied

Idea 1: the spectrum has layers

a: 153 entries



rows: positions j , $j + 51$, $j + 102$



c: 51 entries, each the sum of a column



frequency k of the 153-sequence



 multiples of 3 (17 frequencies): PSD of a equals PSD of c . Decided by c alone.

 multiples of 9: already fixed by the seed.

 the other 51: need the full sequence.

Stage 1: search the folded sequences first

$$1.2 \times 10^{17}$$

folded candidates c consistent with the seed (each seed entry splits into three sub-sums)

17 constraints

PSD of c at $k' = 1, 2, 4, 5, 7, 8, 10, 11, 13, 14, 16, 17, 19, 20, 22, 23, 25$ (PSD(17)
fixed exactly)

to be measured

canonical survivors per side

$$10^4 \times 117$$

stage 1 alone is a network job at 153 (2 hours
on a laptop for 117)

1 complex add

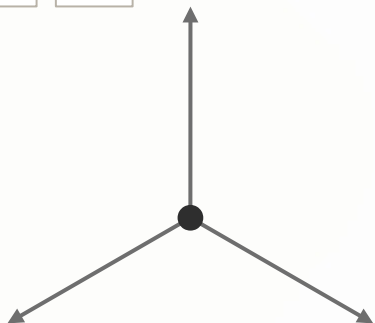
per frequency per candidate, with prefix sums over
blocks

exact

same survivors as an independent check, index
by index

Idea 2: most of the sequence is invisible

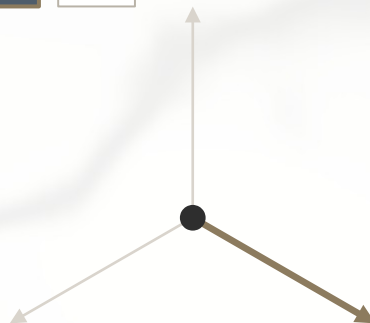
a sub-triple with equal entries ($c = \pm 3$)



$$1 + \zeta + \zeta^2 = 0$$

contributes nothing

a sub-triple with one odd entry ($c = \pm 1$)



$$-2 s \omega^{kp}$$

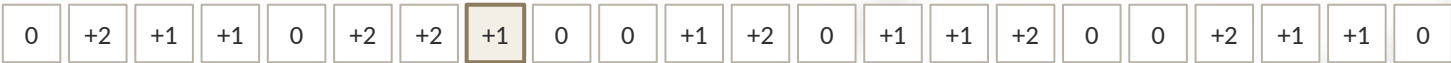
one term, pointing one of three ways

For every frequency not divisible by 3:

- entries of c equal to ± 3 drop out entirely
- each entry equal to ± 1 adds one term, whose direction depends only on which of its three positions holds the odd element
- typically about 40 such entries, so the 153-term sum becomes a 40-term sum

Stage 2: walk the choices without ever starting over

offset chosen for each of the ~40 odd-element columns (0, 1 or 2)

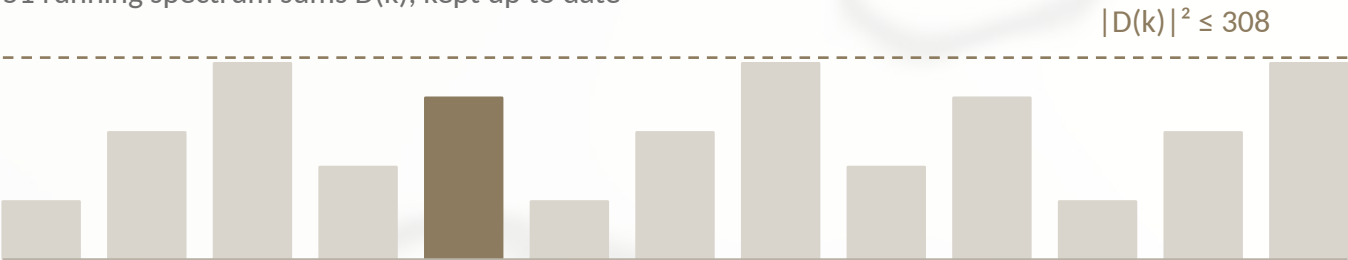


... $3^{38} \approx 10^{18}$

Gray-code order: each step changes exactly one choice by one

full sequences behind one typical survivor

51 running spectrum sums $D(k)$, kept up to date



no branches

one complex add per frequency per step: the same kernel as 117, sized for 51 frequencies

one complex add per frequency per step, then compare against the limit

verified

every emitted sequence re-checked independently:
structure, all PSDs, key

Idea 3: symmetry, search one copy of every 108



... 108 versions of the same candidate; the solid row is the canonical one

- shift by 17 positions (9 ways)
- multiply positions by 35, 52, 86, 103 or 137 (mod 153), or reverse
- all keep the seed, and all keep pairs pairs

So keep one canonical copy per orbit, and compare keys that are themselves canonical.

12×

fewer folded survivors (stage 1)

9×

fewer expansions each (stage 2)

same 108-fold group as for 117

Exact keys instead of fingerprints

A tempting shortcut

666442111921141201657071323181785937773194782986569

last decimal digit of each spectrum value, computed in single precision. A hash: equal strings suggest a pair. Rounding near .5 makes a true pair's A and B strings disagree about 3% of the time.

Now

key: the 76 periodic autocorrelations, exact integers, canonical under the multipliers



the sequence itself is never sent: it is rebuilt from the slice and the state index when a key matches



- Equal keys are a proof, not a hint: side B stores -2 minus its autocorrelations, so equality is the pair condition itself.
- A worker sends 12 bytes per record, sorted: a 64-bit hash of the key and the state's index in its slice. The relay stores 16 bytes (hash, slice, index).

Finding the pair is a join

A records, sorted by key

0f3a...
1c88...
2b41...
4d07...
7e92...
9a10...

equal keys

B records, sorted by key

0a19...
1c88...
3f60...
4d07...
8c33...
9a10...

- Each side's records are produced independently, by different workers at different times, and merged by key.
- A match names two sequences. Apply the multiplier that aligns them and check the autocorrelations directly: a Legendre pair, or not.
- Records arrive sorted and are kept in sorted runs, so the join is one streaming merge over both sides. Coverage is tracked separately, as a ledger of slice numbers in one fixed order.

Checked against the known case: length 45

	direct enumeration	new search, full	new search, symmetry-reduced
A sequences passing	4,528,224	4,528,224	42,988
B sequences passing	2,335,068	2,335,068	22,936
pairs	192 fingerprints	33 orbits, every one verified	the same 33 orbits
sequences examined	6.27×10^9	7.67×10^8	7.4×10^6

The 192 pair fingerprints found by direct enumeration are exactly the 12 multiplier images of the 33 orbits: 192 of 192, none extra.

45 = 9×5 has the same structure as $153 = 9 \times 17$ and a known answer, so it is the permanent regression test: every change to the search must reproduce this table.

Proven on 117

Oct 2026

Blackthorn classifies the Legendre pairs of length 117 with the prescribed compression and multiplier 64: one equivalence class

2 patches

where the published pair sits in the 117 search space, located from its compression and Gray-code position

1 of 1

hash match verified: the distributed search produced the pair in those patches, and the join proved it

- The 117 search ran on DCP with browser and native GPU workers, results posted straight to a relay, and a streaming merge join.
- Searching the two patches holding the published pair produced exactly one key match, re-proved on the sequences: a confirmation of the published pair, not a new discovery.
- The same pipeline, with the seed and frequency count changed, is what runs for 153.

What it costs, and how to make it finite

10^{33}

flat enumeration, per side

→ **$\approx 10^{31}$**

after stage 1 (folded search)

→ **$\approx 10^{29}$**

after symmetry (108×)

stage-2 states per side (estimates by analogy with 117; to be measured)

beyond exhaustive search

about ten million times the 117 space. A volunteer network can sample it, not finish it.

$\approx 10^{13}$ states

if both rows are also fixed by multiplier 118 ($\equiv -1 \pmod{17}$, $\equiv 1 \pmod{9}$), as the 117 solution is by 64: $126 \times 84^8 \approx 3 \times 10^{17}$ raw. Hours of GPU.

Plan: search the symmetric family exhaustively first, which gives a definite answer for it, then sample the full prescribed space with exact coverage.

Projected storage

16 bytes

1 : 270,000

≈ 1 GB

CA\$33

stored per record: 7-byte key hash,
slice number, state index

A states that became records for 117
(B: 1 in 6.6 million); 153 will be
measured

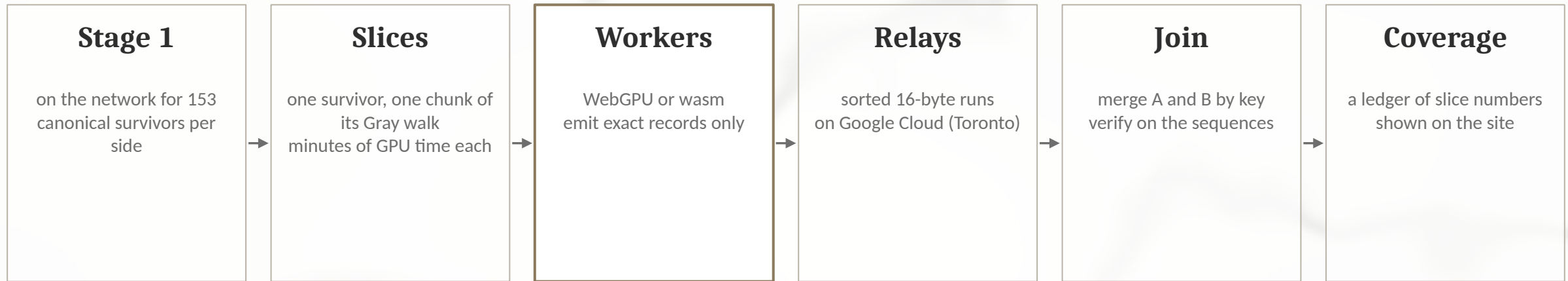
for the whole symmetric family at the
117 record rate (about 10^{13} states)

per TB-month, Cloud Storage
Standard, Toronto

Scenario (estimates, 117 record rates)	states	records	storage
symmetric family, multiplier 118, both sides	$\approx 10^{13}$	$\approx 10^7$ to 10^8	≈ 1 GB
10,000 laptop GPUs for a day, full prescribed space	8.6×10^{16}	$\approx 1.6 \times 10^{11}$	2.6 TB
10,000 data-centre GPUs for a month, full space	2.6×10^{19}	$\approx 5 \times 10^{13}$	790 TB
the whole prescribed space, both sides	$\approx 10^{29}$	$\approx 4 \times 10^{23}$	far beyond any store

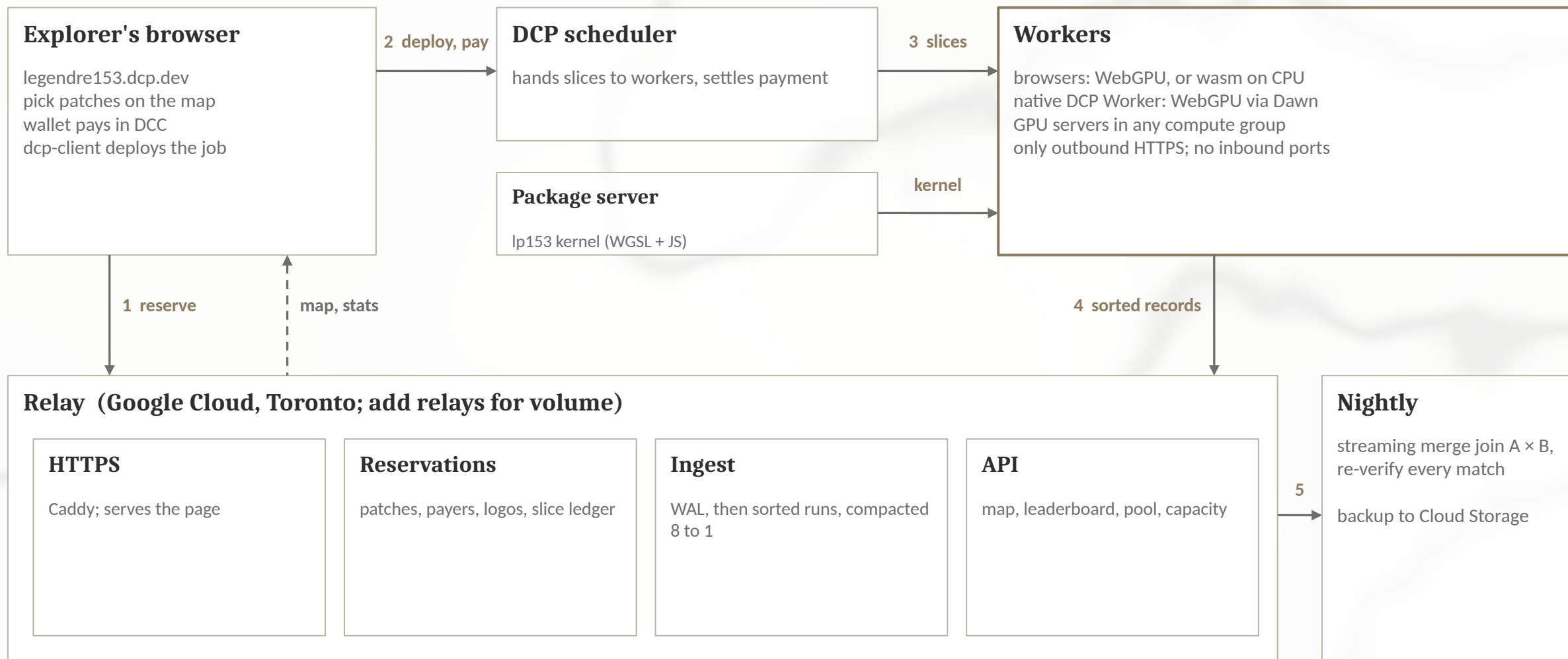
The symmetric family fits on a laptop. The full space is bounded by storage, as it was for 117: coverage is set by a budget, and relays stop taking work when it is reached.

On the network



- A slice is a survivor plus a range of its Gray walk; the walk can be entered at any index, so slices are uniform and restartable.
- Workers return only records, never the search state, and send them straight to a relay over HTTPS, not through the scheduler.
- Anyone can pick a region of the space and run it on their own GPU; the site shows what has been covered.

Architecture



A slice is a few minutes of GPU work. Workers fetch the kernel, walk their slice, sort the records and POST them to the relay named in the job.

Join the hunt

Pick a region of the space. Run it on your GPU. Watch the coverage grow.

