The Bag Has to Fit

A LEGO paper-bag transition becomes a production-system test for Tool Builder: a working replacement is not the same as installed capability.

2026.08.18 22:01 DENNIS HEDEGREEN ANALYSE V1.0
HTTPS://HEDEGREENRESEARCH.COM/ARTICLES/THE-BAG-HAS-TO-FIT/

My son got some LEGO.

The bricks looked normal.

The bags did not.

Instead of the clear plastic pre-pack bags I remember, the loose bricks were inside paper-based bags.

That was enough to create a small question: when did LEGO change them?

At first, this looked like one of those changes that should have a very short history. A company decides it wants less single-use plastic. Somebody finds a suitable paper. The packing machine gets a new roll. Done.

The actual timeline is less tidy.

In September 2020, the LEGO Group announced that it would begin phasing out the single-use plastic bags used for loose bricks inside LEGO boxes. Paper-based bags were planned for trials from 2021.[1]

There is an easy but wrong version of the story from here: LEGO had a paper bag in 2020 and somehow spent the next seven years putting it into boxes.

That is not what the evidence says.

The 2020 announcement already said that moving away from the existing packaging was not a simple task. LEGO had been exploring alternatives and testing prototypes, but the replacement material still had to be durable, lightweight and compatible with the building experience.[1]

A direction existed.

A globally installed production solution did not.

That distinction turned out to be the interesting part.

The bag was inside the factory

In 2023, LEGO described some of what happened next.

The company said work on the production transition had begun in 2022 with designers and engineers trying to produce a paper-based bag that could meet its quality requirements.[2]

The bag had more jobs than I had initially given it credit for.

LEGO's pre-pack bags hold groups of loose bricks. They carry numbers that indicate building sequence. They need to survive shipping. They also have to survive the manufacturing process itself.[2]

Paper introduced another problem: climate.

LEGO said paper was more sensitive than plastic to changes in humidity and weather. A bag therefore had to work in factories operating under different environmental conditions, from hotter and more humid locations to colder ones.[2]

Around 180 different papers and formats were tested.[2]

That number is useful, but not because 180 is inherently extraordinary. It is useful because it reveals what the object really was.

This was not simply a packaging-material decision.

The bag was a component inside a production system.

It had to fit the bricks, but it also had to fit the machines, the process, transport, the building sequence, quality expectations and the conditions inside different factories.

By November 2025, LEGO said 56% of the packing lines that package LEGO bricks had transitioned away from single-use plastic bags. The rollout was largely complete in China and Vietnam. Implementation in the Czech Republic and Hungary was targeted for 2026, with Mexico during 2027.[3]

Even that was not the end of the engineering story.

LEGO described completion of the rollout as the end of a first phase and said it would continue looking for ways to make the paper-based packing lines work faster, with the aim that their output could eventually surpass the plastic packaging machines being replaced.[3]

So there are several different statements that can all sound like `the paper bag works`:

The material can form a bag.

The bag can hold the bricks.

The bag can survive shipping.

A factory can manufacture and fill it.

A production line has been converted to use it.

Enough production lines have been converted to make it normal production.

The new line has recovered or exceeded the performance of the old one.

Those are not the same achievement.

A working replacement is not installed capability

That is the claim I want to keep from this case:

> **A working replacement is not the same as installed capability.**

The distinction is easy to lose when we look at a finished object.

I can hold the paper-based bag in my hand. It works. The existence of that working object makes the remaining problem look retrospective and almost silly.

Why not just use this one everywhere?

Because `this works` answers a different question from `the production system can now do this routinely`.

A component can be ready enough to demonstrate and still not be absorbed by the infrastructure around it.

And absorption is not necessarily the final stage either. LEGO's own account separates rollout from the later effort to improve line speed.[3]

There is a rough sequence here:

`functional → compatible → deployed → migrated → optimized`

I do not mean that as a universal engineering law. Real production transitions are messier than five boxes and arrows.

It is simply a useful guard against collapsing everything into one word: `works`.

A prototype can work.

A single line can work.

A converted factory can work.

A globally migrated production system can work.

They tell us different things.

Fifty-six percent of what?

The number that stayed with me was not 180.

It was 56%.

LEGO could say that 56% of the relevant packing lines had transitioned.[3]

For that statement to mean anything, there has to be a counted population underneath it.

There is some answer to:

> **56% of what?**

That sounds trivial until you try to measure a migration in a system whose boundaries have not been made explicit.

What belongs in the denominator?

Everything that has ever existed?

Only things currently in production?

Things intended to remain in production after the transition?

What about something that will be replaced rather than converted?

What about something deliberately retired?

A migration percentage can look objective while hiding a subjective decision about what counts.

And that was where the paper bag stopped being a LEGO question for me.

My smaller factory

Hedegreen Research has been building a Tool Builder for TID.

The immediate reason is simple. I can ask Codex to build individual tools quickly. But producing each tool as a mostly fresh software problem means paying repeatedly for decisions that should become infrastructure: identity, packages, responsive behavior, routes, receipts, versions and other pieces of production plumbing.

I wrote about that problem in *The Bricks Have to Fit*.[4]

The aim is not to make every tool identical.

It is to make repeated infrastructure stop being a fresh invention every time.

That sounds like a software architecture problem.

The LEGO bag gave me a different way to inspect it.

I had been asking some version of:

> Can Tool Builder make the tool?

But that is equivalent to asking whether the paper bag can hold bricks.

It is necessary.

It is not the whole migration.

A more difficult question is:

> **Has the capability to produce and continue this tool actually moved onto the new production system?**

A legacy tool can be visible in a new interface without being migrated.

A new system can import old output without owning the process that created it.

A tool can even be rebuilt once through a new path while its next revision still depends on the old one-off workflow.

The distinction I care about is production ownership.

Could the new system preserve, rebuild and continue the capability without falling back to the production path it is supposed to replace?

If not, something still depends on the old factory.

What is TID's 56%?

This is the part of the article that is still a test, not a result.

TID has a Tool Build Ledger.[5]

But a ledger existing is not enough.

The current public ledger already shows useful production-state information. It distinguishes live tools, public prototypes, private proofs and Tool Builder-built objects. It records version, receipt and package information for some entries, and it is explicit where detailed version, hash and Tool Builder migration history still need backfill.

That is progress.

It is not yet the same as LEGO's counted migration denominator.

Before I can honestly publish an equivalent of LEGO's `56%`, I need the ledger to answer one larger question:

> **Which capabilities are supposed to survive the move into the new production system?**

That question breaks into smaller ones.

What exactly is being migrated?

What counts as migrated rather than merely imported or registered?

Which old things should be carried forward, replaced, archived or retired?

And when an object leaves the counted population, is that exclusion visible?

That last question matters because migration metrics are easy to flatter.

If inconvenient legacy objects can quietly disappear from the counted population, the percentage can rise without production capability moving very far at all.

So the first requirement is not a percentage.

It is a denominator I am willing to defend.

> **You cannot measure a migration until you know what has to survive it.**

Some things should not cross

There is another trap here.

A serious migration should not become a promise to preserve every historical implementation.

Some old tools may still represent useful capability and should move across.

Some functions may matter while the existing implementation does not, in which case replacement is cleaner than migration.

Some artifacts may deserve to remain readable as history without creating a future production obligation.

Some things should die.

The target is not preservation of every old file.

It is preservation of the capabilities we deliberately decide the future system should still have.

That also prevents the old production system from dictating the architecture of the new one.

If Tool Builder must reproduce every accidental historical decision merely because it once existed, the migration can turn into restoration work.

The future factory becomes a museum of the old factory.

That is not the goal.

Can I turn the old factory off?

The cleanest migration test I found during this work is almost embarrassingly simple:

> **Can the old production path be turned off?**

Not: can all the old files be deleted?

Historical artifacts can remain.

The question is whether current production still requires the old path.

Can a surviving tool be revised?

Can a known-good version be recovered?

Can its package and identity be understood?

Can it move through the current production process without somebody returning to the retired one-off method and rebuilding the missing pieces by hand?

If the answer is no, the legacy path is not yet legacy.

It is still infrastructure.

That gives me a much stronger definition than `the new Builder knows this tool exists`.

Migration is complete only when required capability can continue without the production system being retired.

Again, that is currently a proposed TID test. It is not a claim that HR has already reached that state.

A new factory can still be worse

There is one more uncomfortable consequence.

Suppose every relevant TID tool eventually becomes Builder-native.

That alone would not prove that building Tool Builder was a good decision.

A cleaner architecture can still be a worse production system.

If the new path permanently takes more human work, more AI usage and more bespoke intervention while producing no meaningful gain in reliability, reproducibility or capability, then architectural neatness has become expensive decoration.

LEGO's continuing work on packing-line speed is useful here because it separates conversion from production performance.[3]

For TID, the equivalent comparison cannot be only `how fast did Codex generate the first HTML file?`

The old one-off workflow can look extremely fast if the clock stops the moment a tool first appears.

The real cost continues through validation, deployment, responsive fixes, later revisions, recovery and repeated reinvention of the same infrastructure.

Tool Builder should eventually justify itself somewhere across that lifecycle.

Maybe it uses fewer bespoke AI interventions.

Maybe revisions become cheaper.

Maybe failures become easier to locate.

Maybe packages become reproducible.

Maybe strong tools become easier to build with less access to expensive AI production capacity.

But one of those benefits has to become real enough to observe.

> **The factory is not justified because it exists. It is justified when production becomes better because it exists.**

The ledger should show the factory

That leaves a concrete requirement.

The Build Ledger should eventually be able to show more than whether a tool is public, live or ready.

It should make the production transition inspectable.

Not with a giant dashboard invented for this article.

Not with a vanity percentage chosen before the denominator is stable.

Just enough information to answer the questions that now matter:

What has to survive?

What production path owns it?

What is still dependent on legacy production?

What was deliberately excluded, and why?

Is the new path improving anything that matters?

Some of that work has already begun inside Tool Builder. Versioned TID publication is now a declared requirement. The registry still needs to preserve older public versions while pointing to the current one. A small Hello World tool is supposed to make Builder development visible through safe public updates before larger tools move again.

Those are not side details.

They are the beginning of the denominator.

Then, at some point, TID may deserve its own version of `56%`.

I do not want to invent that number now.

The point of the ledger is to make it possible to earn it later.

> **If migration is the work, make the migration visible.**

Back to the bag

My son did not ask how many factories had been converted.

He did not ask how humidity changed the material requirements or how many paper formats were tested.

He opened the bag.

The bricks fit.

That is what successful infrastructure often looks like at the point of use.

A difficult transition becomes ordinary.

The user sees the new state, not all the machinery required to make it normal.

I am not claiming TID is there yet.

For now, I can still see the factory floor.

That may be exactly what the ledger needs to show.

Not because the factory floor is the product.

Because until the migration is visible, I cannot honestly say how much of the work has moved.

Sources

[1] LEGO Group, 15 September 2020, `LEGO Group to invest up to US$400 million over three years to accelerate sustainability efforts`.

https://www.lego.com/en-us/aboutus/news/2020/september/sustainability (https://www.lego.com/en-us/aboutus/news/2020/september/sustainability)

[2] LEGO Group, 21 November 2023, `More LEGO boxes in Europe and Asia to contain paper-based bags`.

https://www.lego.com/en-us/aboutus/news/2023/november/lego-boxes-in-europe-and-asia-to-contain-paper-based-bags (https://www.lego.com/en-us/aboutus/news/2023/november/lego-boxes-in-europe-and-asia-to-contain-paper-based-bags)

[3] LEGO Group, 26 November 2025, `Over half of LEGO Group packing lines switch from plastic bags to paper-based alternatives`.

Current official locator in candidate source register:

https://www.lego.com/fr-be/aboutus/news/2025/november/over-half-of-lego-group-packing-lines-switch-from-plastic-bags-to-paper-based-alternatives (https://www.lego.com/fr-be/aboutus/news/2025/november/over-half-of-lego-group-packing-lines-switch-from-plastic-bags-to-paper-based-alternatives)

[4] Hedegreen Research, The Bricks Have to Fit.

https://hedegreenresearch.com/articles/the-bricks-have-to-fit/index.html

(https://hedegreenresearch.com/articles/the-bricks-have-to-fit/index.html)

[5] Hedegreen Research, TID Tool Build Ledger.

https://hedegreenresearch.com/tid/tool-build-ledger/tool/index.html

(https://hedegreenresearch.com/tid/tool-build-ledger/tool/index.html)

CANONICAL URL: HTTPS://HEDEGREENRESEARCH.COM/ARTICLES/THE-BAG-HAS-TO-FIT/
PDF VIEW GENERATED: 2026-08-18T20:03:24.816Z