
The hardware question

What happens to a digital signage estate when the software vendor goes away, and what to do with the players you already own.

Scala

X2O Media

RDM

Existing hardware

SUMMARY

Two platforms closed. One changed hands. The screens are still on the wall.

In spring 2026, three digital signage platforms owned by STRATACACHE stopped being the thing their customers bought. X2O Media and RDM wound down and are gone. Scala was sold to Vertiseit, folded into Dise, and is very much still being sold, with an announced change of direction. Those are three different situations, and only the first two are endings.

The software side of that story has been covered thoroughly. The hardware side has not been examined at all, and it is the part sitting in customers' buildings.

A large share of that hardware is Scala players, because STRATACACHE sold the same boxes across the family. They are general-purpose computers, most running a long-term-support edition of Windows, a substantial minority running Ubuntu. They are fine machines. What none of them has is a vendor who is answerable for the operating system on them.

That was never unusual. It is how almost all digital signage works: the hardware comes from one company, the operating system from a second, the signage software from a third, and the customer carries the seams between them. It only becomes visible when one of the three disappears.

They are also old. The player generations still in service shipped between roughly 2016 and 2019, which is visible in the operating systems they left the factory with, and they are documented as current hardware as recently as this year. Seven to ten years on a wall is not a projection about this market; it is the age of the estate as it stands.

This briefing sets out what those players actually run, why nobody can tell you their current state from the outside, and what we think the useful question is. It also covers what happened when we put our own operating system onto one of these players, including the parts that are not finished.

We build and sell digital signage, so we have an obvious interest in this subject. Everything factual here is sourced to public documents and can be checked without reference to us. Where we describe our own testing, we say so and we say what we have not tested.

3

companies involved in a single signage screen: the hardware, the operating system, the signage software

0

of them contractually responsible for keeping that operating system patched

7–10

yrs

age of the player generations still in service, against a support window fixed from the operating system's release date rather than your install date

And how many of these players are there?

Three published figures, none of which quite answers the question, but together they give the order of magnitude.

Scala, in its own press-release boilerplate, carried unchanged across years of announcements:

"Scala's network of partners and developers located in more than 90 countries drives more than 500,000 screens worldwide."

STRATACACHE, on its own website, states group-wide figures of **2.5 million managed devices** and **4 million digital signs globally**, across 30 offices and 1,200 employees.

The gap between those last two numbers is the interesting part. Four million signs against two and a half million devices implies roughly one and a half displays per device, which is what you would expect from an estate where a meaningful share of players drive more than one screen. That is the multi-panel pattern this briefing comes back to later, visible in a group statistic.

What the numbers do not tell you

The STRATACACHE figures are group-wide and span businesses with no connection to Scala, X2O Media or RDM, and they carry no date or breakdown. Scala's figure is a vendor claim rather than an audited one, is several years old, and counts screens its software drove on any hardware rather than players it sold. **X2O Media and RDM never published comparable figures at all.**

So the size of the specific installed base this briefing is about remains unknown, and the share of it running on physical Scala players is unknown too. That absence is worth noticing rather than filling in: a category of this scale has no reliable public account of its own installed base, which is precisely the condition that makes a vendor's disappearance difficult to plan around.

The question this briefing is about

Every vendor in this market will tell a customer they can keep their existing hardware. Almost all of them mean their software will install onto the operating system that is already there, frequently as a desktop application, often a Chromium-based one, running as one window among many on a machine that is still doing everything else it was doing before.

So the question worth asking any of us is simple: what happens to the operating system on those players, and who is responsible for it afterwards?

A timeline, from public sources

WHEN	WHAT
March 2026	Layoffs across STRATACACHE, reported against tariffs, supply chain disruption and memory prices.
~2 April 2026	X2O Media winds down after twenty years, reported as entering administration. No transition plan, no support organisation, no roadmap. Up to fifty jobs affected. Montreal-based; founded 2006; owned by Barco from 2014 and STRATACACHE from 2018.
~2 April 2026	RDM (Real Digital Media) winds down the same week, also after two decades. Its CEO retires as operations close.
21 May 2026	Scala sold to Vertiseit for SEK 265 million and folded into Dise. Founded 1987. Reported at the sale at approximately SEK 200 million annual revenue against SEK 85 million of recurring revenue, with more than 1,000 customers and more than 100 partners.

What Vertiseit has said about Scala's direction

Quoted rather than characterised, because their own words are clearer than anyone's interpretation of them. Vertiseit intends to

accelerate Scala's transition toward a modern SaaS-based and device-agnostic offering, fully aligned with its partner-first go-to-market strategy

Vertiseit, acquisition announcement, May 2026

with focus areas of “accelerated product innovation, strengthened partner enablement, expanded hardware flexibility, and a gradual transition from traditional perpetual licensing toward a true SaaS offering.”

Read as a customer rather than as an investor

Three of those points describe changes to the arrangement an existing Scala customer bought into.

- **A repricing is coming, and it is a decision trigger rather than a grievance.** Scala's reported figures put roughly SEK 200 million of revenue against SEK 85 million of recurring revenue, so a substantial share of the base paid once rather than annually, and reporting on the plan puts two to three years on converting them. We should be straight here: Screenly is subscription-only, so we are in no position to argue that subscriptions are worse than perpetual licences. The point is narrower and more useful. A recurring cost is arriving that was not in the contract these organisations signed, and at the moment a platform starts costing money again, "is this the right platform?" stops being a hypothetical question.
- **The product is going device-agnostic, and the hardware business with it.** Dise's own headline describes a "hardware-agnostic platform". Reporting on the Scala plan is more direct: "the hardware business is expected to be exited rapidly, either through discontinuation or transfer to partners."
- **Direct customers cannot stay direct.** This is not an inference from a press release. Dise states it on its own website: "Dise is exclusively delivered through certified digital in-store integrators. We do not sell direct. We do not compete with our channel." An organisation with a direct Scala contract, direct support and a named account manager has nowhere direct to go.
- **And the platform is now retail-only.** Dise describes itself as an in-store experience management platform "built for global retail scale", and the verticals it names are retail chains, grocery and supermarkets, fashion and apparel, pharmacy and health retail, convenience and fuel, department stores, auto dealerships and franchise networks. Every one of them is retail.

Scala's installed base is not retail-only. It never was: banks, transit hubs, cinemas, corporate campuses, universities, hospitals and venues all ran it, and Scala's own EMEA leadership described its UK footprint as "iconic landmarks as well as transport hubs, banks and high-end retailers". Three of those four are not stores.

None of this is hidden or unusual. Dise has been a retail platform since 2017 and says so plainly, and a partner-only model is a legitimate way to run a software business. The issue is only that a large part of Scala's installed base does not match any of it.

The part that matters if you own the players

Scala's hardware was built inside STRATACACHE, and STRATACACHE is a hardware company. Vertiseit and its subsidiaries are software companies. So the acquisition separated the software from the manufacturer that made the players, and the hardware business is reported as being exited rapidly, by discontinuation or by handing it to partners.

Whatever happens to the software, those players now have no vendor building them. That is not a prediction. It is the structure of the deal.

One thing we will not claim

Scala has not been discontinued. It has a new owner with a stated direction, which is a different thing, and anyone telling a Scala customer otherwise is worth treating carefully. X2O Media and RDM are a different case: those platforms closed, and their customers have no acquirer to wait for.

What those players actually run, and why nobody can tell you

Assembled from Scala's own product documentation, version 13.60. As far as we can establish, this table has not been published anywhere else. Read the column heading carefully: this is what each model *shipped* with.

MODEL	SHIPPED OPERATING SYSTEM	SUPPORT ENDS, IF STILL ON THAT BUILD
Player-L (B308)	Windows 10 IoT Enterprise 2016 LTSC	13 October 2026
Player-DX (B396)	Windows 10 IoT Ent. 2019 LTSC, or Xubuntu 20.04	9 January 2029, or Ubuntu window
Player-Q (B396)	Windows 10 IoT Enterprise 2019 LTSC	9 January 2029
Player-S (B409, discontinued)	Windows 10 IoT Enterprise 2019 LTSC	9 January 2029
Player-SP (B409, discontinued)	Windows 10 IoT Enterprise 2019 LTSC	9 January 2029
Player-S (R1305G)	Windows 10 IoT Ent. 2019 LTSC, or Ubuntu 22.04	9 January 2029, or Ubuntu window
Player-SP (R1505G)	Windows 10 IoT Ent. 2019 LTSC, or Ubuntu 22.04	9 January 2029, or Ubuntu window
Player-NX4	Ubuntu 20.04 only	Ubuntu window
LINQ Tablet / Shelf Edge	Android	—

Four things this table is commonly misread as saying

- **It is not consumer Windows 10.** These are long-term-support editions with their own lifecycles, and most commentary on this subject has conflated the two.
- **A large part of the line ran Linux.** Player-DX, Player-S, Player-SP and Player-NX4 all shipped in Ubuntu or Xubuntu variants. Any analysis of this installed base that assumes Windows throughout is wrong for a significant share of it.
- **The shipped build depends on purchase year, not just on model.** A later unit means a later build with a later end date, so even the baseline varies within one model name.
- **Most importantly: what shipped is not what is installed.** These fleets are five to ten years old, and reimaging a fleet is ordinary practice in any organisation with a managed desktop estate. So the build a model left the factory with says very little about the build running on the wall today, and the gap grows with the age of the deployment.

The same boxes turn up under a different brand

RDM sold media players too, under the NEOCAST name, and the model list is worth putting next to the one above.

RDM NEOCAST MODEL	HOW RDM DESCRIBED IT	SILICON THE NAME POINTS TO
NEOCAST Z8350	Entry level, ultra-compact, fanless	Intel Atom x5-Z8350
NEOCAST R1305G	Performance, multi-zone video, audio and HTML5	AMD Ryzen Embedded R1305G — the same part as Scala Player-S
NEOCAST R1505G	Performance, dual channel, multi-zone	AMD Ryzen Embedded R1505G — the same part as Scala Player-SP
NEOCAST V1202	Dual 4K UHD output	AMD Ryzen Embedded V1000-series
NEOCAST V1202-Q	Quad HD output	AMD Ryzen Embedded V1000-series

Two of those model names are AMD part numbers, and they are the same part numbers Scala's own documentation gives for Player-S and Player-SP. Which is to say that STRATACACHE was shipping a common hardware platform across its companies with different badges on the front, and the naming makes it obvious rather than hidden.

It matters for anyone holding this hardware. A NEOCAST R1505G and a Scala Player-SP are, on the evidence of their own product naming, the same machine sold by two subsidiaries of the same group, one of which has closed and the other of which has been sold to a company exiting the hardware business. And **RDM never published what operating system any of these players ran**, so for that estate even the starting point in the table above does not exist.

The finding underneath the table

Nobody can tell you what several hundred thousand signage screens are currently running. Not their software vendor, not us, and in a good number of cases not the organisation that owns them.

That is not a gap in the research. It is the more interesting result, and it says something about how signage estates get managed that has nothing to do with who owns Scala. The only reliable way to find out is to ask whoever holds the imaging process, which is a different conversation from the one this industry has been having.

The one lifecycle point that does hold, whatever the build

There is no supported in-place upgrade path from a Windows 10 long-term-support build to Windows 11 IoT Enterprise LTSC. And for a large part of this hardware there is no supported destination either.

Microsoft's published list of AMD processors supported for Windows 11 IoT Enterprise LTSC 2024, 24H2 and 25H2, updated in June 2026, covers the Ryzen Embedded 8000 and 7000 series, the **R2000** series, and the

V3000 and V2000 series. It does not include the **R1000** series or the **V1000** series.

Those are the parts in question. Scala's Player-S is an R1305G and its Player-SP an R1505G, both R1000-series; RDM's NEOCAST V1202 and V1202-Q are V1000-series. Microsoft supports the generation that came after these chips, and not these chips. So whenever one of these fleets moves to a supported Windows, it is a wipe, a reimage, an application revalidation and, on a good deal of this hardware, new machines as well. That is worth knowing not as a deadline but as a cost comparison: **the work involved in moving to a different platform is not larger than the work involved in staying on this one.**

Who is responsible for the operating system?

This is the part of the story that outlasts the news, and it applies to almost every signage deployment in the world, not only to the ones affected this spring.

Three parties build every screen, as set out at the start of this briefing, and each of them does its part competently. The gap is not in any of their work. It is in the space between them, and the operating system is where that space sits.

It is nobody's product. The hardware maker did not write it. The signage vendor did not write it and cannot patch it. The operating system vendor has no

idea the machine is showing a menu board. Every party can point at another one, and all of them are right.

Scala's own product documentation states the position plainly for its Windows players, and it is worth quoting because it is entirely typical rather than unusual.

Windows does not provide feature updates nor revisions for IoT Enterprise LTSC operating systems, however security patches are provided if your players are appropriately connected to allow for patches.

Scala product documentation, v13.60, player configuration

Read that as an operations statement rather than a disclaimer. Keeping the operating system current was a function of the customer's connectivity and configuration. It was never the signage vendor's to answer for, and the same is true across the category.

It is not only a Windows problem

The pattern repeats wherever the operating system belongs to someone else. A smart display running Tizen or webOS inherits a support window set by television product cycles. An Android player stops receiving updates when the device maker discontinues the SKU. A signage box running Ubuntu depends on somebody in the organisation owning that lifecycle, which in practice often means nobody does.

What this costs, in practice

Whoever owns a signage estate today owns a fleet of general-purpose computers. Patching, and testing the patches. Imaging and reimaging. Drivers. An operating system lifecycle set by a vendor with no interest in signage. Security exposure that an auditor will eventually find. And a site visit every time one of them stops behaving.

None of that appears in a signage budget, and all of it appears in somebody's week.

Three layers, one vendor

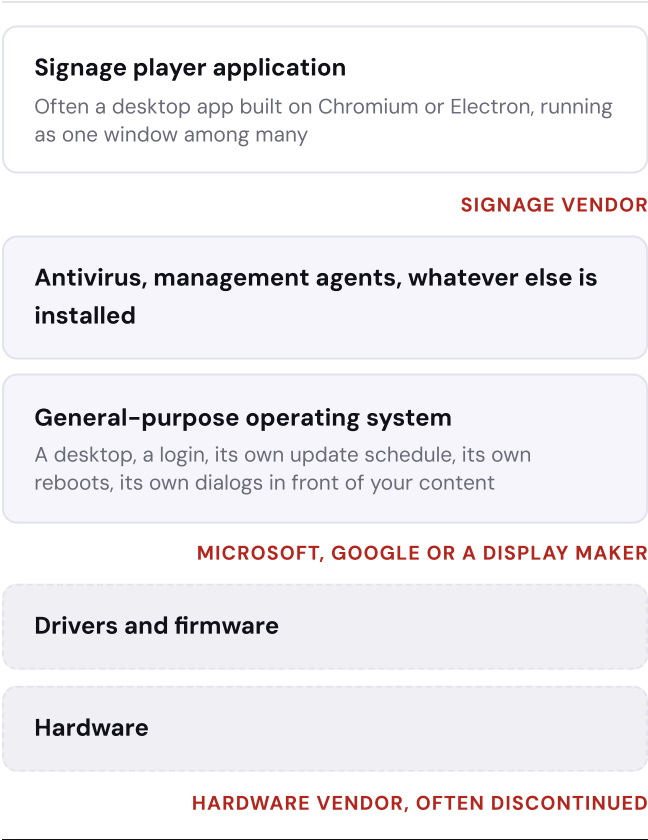
Digital signage is three layers. Most vendors own one of them. We build all three and release them as a single product, which is the only arrangement under which anyone can be answerable for the whole device.

LAYER	WHAT IT COVERS	WHO OWNS IT, TYPICALLY	SCREENLY
Content	Authoring, scheduling, publishing	The signage software vendor	YES
Device management	Monitoring, control, provisioning, diagnostics	A separate management vendor, or nobody	YES
Operating system and device	Boot, drivers, patching, hardening, lifecycle	Microsoft, Google, or a display manufacturer	YES

What that looks like on an actual machine

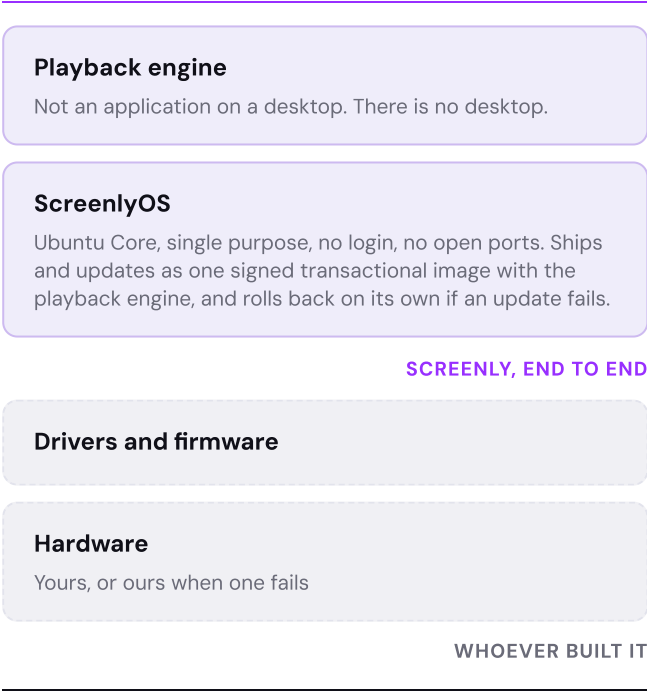
The difference is easier to see as a stack than as an argument. On the left, a typical signage deployment on hardware the customer already owns. On the right, the same hardware after a reimage.

SIGNAGE SOFTWARE ON AN EXISTING MACHINE



Three vendors. Nobody owns the device.

THE SAME HARDWARE, REIMAGED



One vendor. One update stream. One SLA.

This is not a criticism of building on a general-purpose operating system; it is how nearly all signage software has to work, because nearly no signage vendor has an operating system of its own. It is a description of where responsibility ends up, which is with whoever owns the estate.

ScreenlyOS, where we can own the device

Our own operating system, built on Ubuntu Core in partnership with Canonical, and hardened down to a single job. The device does nothing else, which is what makes the rest possible.

- **Nothing to patch.** The operating system and the playback software update together, over the air, cryptographically signed and transactional, rolling back automatically if an update fails.
- **Closed by default.** No remote login. No open ports. Encrypted storage where the hardware supports it.
- **Certified.** SOC 2 Type II and ISO 27001.
- **Built for enterprise networks.** 802.1x on wired and wireless including static IP from the on-screen setup wizard, mutual TLS client certificates so a screen can authenticate to internal services, SAML single sign-on, audit logs.
- **Manageable without site visits.** Remote reboot, remote diagnostic screenshots, and CPU, memory and network health from the dashboard.
- **Keeps playing offline.** Content and schedules are both cached on the device, so day-parting continues on time when the connection drops.
- **Runs on hardware we did not build.** Our own players, Raspberry Pi, Sharp displays with it preloaded, generic x86, and Scala players.

Screenly Anywhere, where we cannot

No real estate is uniform. Where the device cannot take our operating system, Screenly runs as an application on top of the one it already has: Samsung Tizen, LG webOS, Android and Android TV, the Amazon Signage Stick, ChromeOS Flex, or a plain browser. Same cloud, same dashboard, same content, schedules and labels, and it appears alongside every other screen.

And the honest difference between the two

On Anywhere we manage what happens inside our application, but not the system around it, so updates and security depend partly on the host. The host operating system is still doing its own job: it can install its own updates and reboot, compete for memory, or put its own prompts in front of your content. On some platforms apps render on our side and are sent to the screen as images rather than running on the device.

None of that happens on ScreenlyOS, because the device does nothing else. **For the deepest control and the strictest security, ScreenlyOS. For reach and flexibility, Anywhere. The same dashboard either way.**

A note on language, since it matters in this market. Screenly is not device-agnostic. Anywhere is an option for hardware we cannot own, not a description of how we are built. Where we can own the device we do, because that is the only way anybody can be accountable for the operating system on it. We also do not sell these layers separately: a customer cannot take our operating system and run it under a different content management system, and there is no plan to offer that. We sell a complete device, not a component.

What happened when we put our operating system on a Scala player

We wanted to know whether Scala hardware has a working life after Scala, so rather than reason about it we went and found some. What follows includes the parts that did not work and the parts we have not attempted.

We could not buy the model we wanted

We set out to test the Player-Q, the four-output model, and spent several weeks trying to source one. That included approaching contacts inside Scala before the sale went through. We could not obtain a single unit at any price.

Worth recording as a data point in its own right, and note the timing: this was already the position while Scala was still a going concern under its previous owner. Player-S and Player-SP in the B409 generation are documented by Scala itself as discontinued, and the practical availability of the rest is now whatever the second-hand market decides.

The deal structure makes it structural rather than temporary. The players were built inside STRATACACHE, which is a hardware manufacturer; the software went to Vertiseit, whose businesses are software. Reporting on the plan states that the hardware business is expected to be exited rapidly, by discontinuation or by transfer to partners. So for an organisation whose next hardware failure has to be replaced, the question is not whether stock is tight this quarter. It is that nobody is building these machines any more.

What we did test

One unit: a Scala **Player-SP**, the two-output sibling of the model we could not buy. Reimaged with ScreenlyOS, it ran a full workload of the kind these deployments actually carry:

- Applications rendering live data parsed from an external XML feed, running on the device itself rather than server-side
- Video playing inside those applications
- **Two displays driven as a single joined canvas** across its two HDMI outputs, with layouts arranged across the full surface
- Registered and managed in the same dashboard as any other Screenly device

What is not finished

- **Multi-display setup is manual.** Physical arrangement, output mapping, orientation and bezel compensation are all configured by hand. The self-service version of that is not released and we are not giving a date for it.
- **The dashboard's remote preview does not render joined displays correctly.** Remote preview is how an operator checks a screen without visiting it, so this is a real gap and not only a demonstration problem.
- **Every other model in the table on page 5 is untested by us.** Including, specifically, the four-output Player-Q. Two outputs working is a reasonable basis for expecting four, and it is not the same as having done it.
- **We have never booted an X2O-branded or RDM-branded player.** Worth noting what that does and does not mean. RDM's NEOCAST R1505G is named after the same AMD part as the Scala Player-SP we did test, so there is a reasonable expectation it behaves the same way. A reasonable expectation is not a test result, and we will not list a model as supported until somebody has booted one.

Why we publish the untested list

Our compatibility position names the models we have booted and the ones we have not, and states the limitations of both. A list that only contains successes is not information, and anyone evaluating platforms at this scale can tell the difference.

And what happens when one of them dies

This is the question a reimage raises rather than answers, and it deserves a direct answer, because the honest position throughout this briefing is that nobody is manufacturing these players any more.

We build our own hardware, and it runs the same operating system. So a fleet part-way through its life can be part reimaged Scala players and part Screenly players, and the two are not managed differently: same image, same update stream, same dashboard, same behaviour. When a Player-SP finally fails, the replacement is one order rather than a platform decision, and the screen it was driving does not need to be treated as a special case afterwards.

The same is true in the other direction. ScreenlyOS also ships preloaded on Sharp large-format displays with the computer built into the panel, and runs on Raspberry Pi and on generic x86, so an estate can move toward whatever form factor suits each site without the software changing underneath it.

To be clear about what this is and is not

Buying our hardware is not a condition of anything. The image for existing players is included with the subscription and always will be, and an organisation that reimages three hundred Scala players and never buys a single Screenly player is a perfectly good outcome for us.

The point is narrower: **a discontinued player stops being a procurement problem.** Right now, an organisation running this hardware has an open question about what it does the third or fourth time a box fails, and the available answers are the second-hand market or an unplanned platform migration under pressure. Replacing one unit at a time, on the same operating system, is a better answer than either, and it is worth having in place before the failures start rather than after.

Four differences worth knowing before comparing quotes

No hardware purchase to approve

The image goes onto players that are already installed, so there is no capital request, no procurement cycle and no depreciation schedule. It is included with the subscription in the same way our Raspberry Pi image is, and it is never a separate line item.

What that changes over a fleet's remaining life is the shape of the spend. Instead of a refresh that has to be budgeted, approved and executed as a project, replacement becomes one unit at a time as machines actually fail, on hardware that runs the same operating system as the ones still working. Section 5 covers how that works in practice.

Billing is per device, not per display

A billable screen is a device running our operating system, so a multi-panel board driven as one continuous picture from a single player counts once, whatever the panel count.

That is a genuine structural difference from how these estates were licensed before. Scala's documentation sets out single, dual and quad channel licences as separate SKUs across three tiers, recommends that "players with four outputs should typically be assigned quad (four-channel) licenses", and describes the upgrade as "the extra expense" over a single-channel licence. We make no claim about what any particular organisation paid, because that pricing was quoted per deployment. The point is only that the models are different, and that anyone comparing them should compare devices rather than panels.

Pricing is published

Per screen, on our website, with the plan tiers and the feature differences visible without a conversation.

Estates are set up programmatically

Our API is the same one our dashboard and our devices run on, rather than a secondary public interface, with a command-line tool and a Model Context Protocol server on top of it. We have stood up a five-hundred-screen environment through it, including every screen, label, playlist and variable, without anyone entering data by hand. For a migration where the outgoing platform has no clean export format, that difference matters more than any import wizard.

Extending the life of a fleet, rather than describing it as extended

Reusing a working player avoids manufacturing a replacement and avoids disposing of a functioning machine. For an organisation with sustainability commitments, and increasingly with sustainability reporting obligations, that is a reportable outcome rather than a sentiment.

But the version of this argument worth examining is sharper, and it applies to every vendor in this market including us.

Every vendor will tell you that you can keep your hardware. Almost none of them actually saves it, because the thing that forces the refresh is not the signage software. It is the operating system on the box.

Install a new content management system onto the machine that is already there and nothing about that machine's lifecycle has changed. When its operating system reaches the end of its support window, and the next version will not install on its processor, the hardware still goes into a skip. The signage software was never what was holding it back.

Replacing the operating system is the only version of hardware reuse that actually extends the life of the machine. That is a claim about architecture rather than intent, and it is checkable.

We have deliberately not attached a carbon figure to this. Doing it properly means published methodology and stated assumptions rather than a number in a brochure, and that work is not finished.

Where Screenly is not the right answer

Every platform in this category is good at some things and absent from others. Here is our list, so that an evaluation can rule us out early rather than eight weeks in.

IF YOU NEED	OUR POSITION
A drag-and-drop design tool	We do not have one and we are not building one. Content that needs logic or live data is built as an application that runs on the device. That suits organisations with a technical counterpart and does not suit a content team editing templates every morning with no developer.
On-premises content management	No self-hosted version exists. We are cloud-managed. If content management has to run inside your own perimeter, we are the wrong choice, and given recent events that is a reasonable thing to want.
Wayfinding	No application for it today.
Room booking	We can display a room's schedule. We cannot book it.
Delivery to desktops and phones	We do screens. Employee desktop and mobile notification is a different product category.
Immersive meeting rooms	Tiled walls of remote participants, perspective cameras, spatial audio. Not something we do or intend to.
Playlists that fill themselves from content metadata	We have labels and scheduling rules, which cover part of it. Metadata-driven automatic playlists and per-asset expiry dates we do not have.
Rotation across campaign buckets	Pulling one item from each of several campaign groups in a fixed order, as advertising contracts often require, is not something we do today.
Four independently scheduled channels from one player	We drive multiple displays as one joined picture, which is a different model and a better fit for a single continuous board. Four unrelated content streams from one box is not how we work.
Components rather than a platform	We do not sell our operating system to run underneath a different content management system. If you are assembling best-of-breed layers, that is a different kind of vendor.
Broadcast-grade live video routing	We play RTSP and RTSPS streams and camera capture. Trading-floor broadcast integration is not our product.

If most of an estate is on that list, another platform will serve it better, and we would rather say so at the start.

Assessment first, and the old platform stays running

Nothing here requires a decision up front. The parts most likely to make a migration fail are deliberately placed at the beginning, where they cost a conversation instead of a quarter.

STAGE	TYPICALLY	WHAT IT PRODUCES
Stabilisation review Where the platform has closed	1 week	What in the current deployment is frozen, what breaks next, which integrations depend on systems that will keep changing, and what happens if the machine running it fails. Delivered as a document, with no migration proposal attached.
Assessment	1–2 weeks	Every player model checked against what we have actually booted. Every content item sorted into four groups: moves directly, gets rebuilt as an application, gets flattened to a fixed image or video, or does not come across at all. Plus a network and security review. You get the four numbers in writing.
Pilot	2–3 weeks	Up to ten players reimaged and running real content, on spare or decommissioned units. The reimage wipes the player, so we insist on this rather than starting with live screens.
Rollout	~4 weeks per 200 screens	Site by site, lowest-risk sites first. Anyone on site can boot a player from a USB stick, or we work with your existing imaging process. Pairing, playlists, orientation and verification all happen from the dashboard.

Two things we say before anyone commits

Your existing platform keeps running throughout. Because the reimage happens one player at a time, the old system stays live everywhere we have not touched. There is no cutover weekend and no big-bang date. We also recommend keeping the outgoing licence alive for one renewal period past the final site.

If the assessment says do not migrate, we will say so. If too much of an estate lands in the “does not come across” group, the honest answer is that another platform fits better, and the assessment is useful to an organisation that then stays where it is.

Where imaging infrastructure already exists, use it

Large estates are frequently managed by network boot and out-of-band tooling, often because the same team already runs point-of-sale or desktop fleets that way. Where that is in place, a signage migration becomes a scripted operation across hundreds of sites rather than a travel schedule, and nobody visits a screen. Worth establishing early, because the answer moves both timeline and cost more than any other single factor, and it is rarely the first thing a signage vendor asks about.

A free assessment, with no commitment attached

What you get

- Your fleet checked model by model against hardware we have actually tested
- Your content inventoried into the four groups, with counts
- A network and security review covering authentication, proxies and firewall requirements
- A written statement of what moves and what does not
- A rollout estimate

What it costs

Nothing, and there is no credit card and no commitment. It is a document, and it is useful even to an organisation that reads it and decides to stay where it is for another year.

Then, if it makes sense

Up to ten players reimaged and running for thirty days, also at no charge, on spare or decommissioned hardware.

The question to take into any evaluation

“When you say I can keep my hardware, what happens to the operating system on it, and who is responsible for it afterwards?”

Ask it of us as well. It is the question this whole briefing is about, and the answer separates the vendors in this market more cleanly than any feature comparison.

screenly.io · Screenly builds ScreenlyOS, a purpose-built operating system for digital signage based on Ubuntu Core, together with the cloud platform and players that run it. SOC 2 Type II and ISO 27001 certified.

SOURCES

Everything factual in this briefing, and where it came from

Sections 1 to 3 can be verified without reference to us. Section 5 describes our own testing and should be read as a vendor account; we are happy to demonstrate it or share the test notes.

The acquisition and the wind-downs

- Vertiseit, “Vertiseit acquires Scala”, acquisition announcement, May 2026
- Scala, the same announcement
- invidis / Sixteen:Nine, “RDM and X2O Media wind down after two decades in digital signage”, 2 April 2026
- invidis, “Rebuilding Scala: Vertiseit's Toughest Test Yet”, May 2026, for the reported plan to exit the hardware business and the two-to-three-year perpetual-to-subscription conversion timeline
- invidis, “Software M&A: Vertiseit Acquires Scala”, May 2026
- dise.com, for the platform positioning, the named retail verticals, and the partner-only statement “We do not sell direct. We do not compete with our channel.”
- AV Magazine, “Scala opens digital signage showroom in central London”, July 2025, for the description of Scala's UK footprint
- AV Magazine, “X2O Media closes its doors after 20 years of trading”, April 2026
- AV Magazine, “Digital signage icon Scala sold to Vertiseit by Stratacache”, May 2026, for the partner and customer counts
- Scala “About Scala” press-release boilerplate, as carried on PR Newswire releases including the Scala Enterprise 11.07 announcement of 11 April 2019, for the 500,000 screens and 90 countries figures
- STRATACACHE corporate website, stratacache.com, for the group-wide figures of 1,200 employees, 26 languages, 2.5 million managed devices, 30 offices and 4 million digital signs globally. Published without date or company breakdown.
- DailyDOOH, on X2O Media
- Retail Technology Innovation Hub and Retail Optimiser, on the deal terms

The hardware and licensing detail

- Scala product documentation, version 13.60: individual player model pages, “Understanding Player Licenses”, and player configuration guidance
- Microsoft product lifecycle documentation for Windows 10 IoT Enterprise LTSC and LTSC releases
- Microsoft Learn, “Supported AMD Processors for Windows 11 IoT Enterprise (2024 to present)”, updated 24 June 2026, for the supported-processor families
- Canonical published support windows for Ubuntu 20.04 LTS and 22.04 LTS
- realdigitalmedia.com media player pages, for the NEOCAST model list and descriptions
- AMD published product information for the Ryzen Embedded R1000 and V1000 series, and Intel for the Atom x5-Z8350, for the parts the model names refer to

Screenly

- Screenly product documentation on ScreenlyOS and Screenly Anywhere, including the published comparison of what each gives up
- Screenly published pricing
- Screenly platform testing on Scala Player-SP hardware, August 2026

Screenly builds and sells digital signage software and hardware and therefore has a commercial interest in this subject. We have tried to make that interest irrelevant to the factual sections by sourcing them entirely to public documents, and to be specific about the limits of our own testing rather than describing capability in general terms. Corrections are welcome and will be published.

Product names and trademarks referenced belong to their respective owners and are used to identify the hardware and software described. This document does not imply any endorsement or affiliation.