



Hard-hitting questions and answers

We've been picking up on the hard-hitting questions you've been asking us over the years, and it's time to put the record straight.

We hope this document helps ...

Q. What's Java Verified all about?

A. Since 2004 it's been a programme that endorses Java ME apps which pass the relevant tests laid out in the **Java Verified Unified Testing Criteria**.

To gain the endorsement, developers take their Java ME apps through a recognised testing, signing and verification process through the **Java Verified Submission Portal**.

Once an app has gained the Java Verified endorsement, it's then clear to consumers, operators and device manufacturers alike that the app has been tested, and meets the industry standard. The digital signature consequently associated with that app means that the installation of the app onto the appropriate devices will work: the requested permissions will be granted, and the installation prompts will disappear.

Q. Java Verified is owned and run by Oracle (Sun), so it's not an 'industry initiative' at all.

A. Oh yes it is: it's a programme that's now part of the **Unified Testing Initiative (UTI)**: a body set-up to develop and promote app quality for the benefit of the industry. The other programme within UTI is JATAF, which looks at java fragmentation. There's plenty scope for other testing-related programmes to evolve too.

To explain more - since May 2009, Oracle and UTI had been working very closely together to complete the evolution of UTI as an independent industry initiative.

On June 14th that process was completed with the handing over of programme responsibility to UTI.

Oracle sits on the board of UTI (together with Orange, Motorola, Vodafone, AT&T, Sony Ericsson, Samsung, LG and Nokia) but UTI now has an independent status. Therefore, Java Verified and JATAF have an independent status too.

Oracle owns the Java brand, so obviously has an interest in protecting and developing that brand, hence the creation of Java Verified and JATAF in the first place.

Q. Hmm. I still don't get the UTI bit. Please explain more.

A. The Unified Testing Initiative is a non-for-profit body, through which all of the work relating to the various testing programmes is carried out. The testing programmes that currently sit within UTI are Java Verified and JATAF.

|

Q. And is it all about Java ME?

A. We're working on developing other testing-related programmes / initiatives too. Watch this space! Our scope is to work together as an industry, to 'raise the bar' when it comes to the quality of apps.

Q. But whoever it's owned by, it's a money-making machine for them, isn't it?

A. It's not. It's a non-for-profit industry initiative run **by** the industry **for** the industry. AT&T, LG, Motorola, Nokia, Oracle, Orange, Samsung and Vodafone fund Java Verified. No one else contributes.

The costs developers pay to go through the Java Verified process go straight to the test houses (direct) and the people running the signing portal. This 'income' goes nowhere near the Java Verified programme.

For example, a developer purchases a **Publisher ID** straight from TC TrustCenter (the guys who run the Java Verified Submission Portal).

The developer pays for the testing of his app(s) straight to the test house with whom they have the relationship. Even the cost of the testing is negotiated between the developer and his chosen test house. Java Verified has nothing to do with this.

The Java Verified programme takes none of this money.

Q. There's a pot of money just sitting there doing nothing.

A. There isn't. The funds from the members (AT&T, LG, Motorola, Nokia, Oracle, Orange, Samsung and Vodafone) pay for the administration of the programme, the website, the signing portal and any activities dedicated to promoting use of the programme. The members of UTI are responsible for ensuring they're happy with how their companies' money is being spent, and regular accounts are available to the members to review at any time.

Q. But I don't understand what the Java Verified team does: it's all faceless.

A. The members develop the **Java Verified Unified Testing Criteria** against which developers can test their apps. They also constantly look at introducing new ways of making testing easier and cheaper for developers. They develop white papers and documents to make things as clear as possible, and pass on hints and tips based on what they're finding in the daily working lives. This is all done voluntarily. It's done by a core group of people from the Java Verified membership. And yes, the people who answer questions submitted using the form in the 'Talk to us' section of the website are these same people. Real people!

Q. The Java Verified Unified Testing Criteria? Err - what's that exactly?

A. It is the minimum quality requirement which your mobile application must fill. You should not really distribute your application if it does not fill the requirements.

It's a common set of tests for Java ME Technology apps, created and continually revised by members of Java Verified (AT&T, LG, Motorola, Nokia, Oracle, Orange, Samsung and Vodafone) against which apps need to be tested to meet the Java Verified quality standard.

The criteria from technical functionality point of view are organized into categories, which include: **Application Launch, Functionality, Operation, User Interface, Security, Network and Localization.**

A set of test houses has been approved by Java Verified to test developers' apps against the criteria and 'pass' or 'fail' the apps accordingly.

Q. I'm a developer. How do I get the Java Verified endorsement?

A. In a nutshell, you go into the Java Verified **Submission Portal** and purchase a one-off **Publisher ID** for your organisation.

The Publisher ID is a tamper-proof digital certificate that identifies the origin of your applications and authenticates the identity of the person or company providing it – ie you! It'll cost you \$200 and it lasts a year.

You then submit your app(s) JAD and JAR files via the submission portal, select your preferred testing house, contact them direct, and get your app(s) tested. (Use our testing calculator to work out approximately how much it'll cost to get your app tested).

You also have the opportunity of using an R&D signing service via the **Submission Portal**. The **R&D Signature** is a signature given to your application by the portal that allows you to test your apps during the development process yourself, to ensure that they will behave properly and as expected on a 'live' device. Obviously, you can then correct functionality issues prior to finalising your app.

Once your apps have passed the relevant tests with the test house, you'll be notified by email.

The apps will then be signed through the **Submission Portal** with the **Java Verified Signature**. The cost of signing is included in the cost of application testing.

It's as simple as that. As long as your apps pass the criteria, there'll be no problem.

Q. Do we really need Java Verified?

A. Well, yes. The Java Verified process described above was originally designed to streamline testing and signing procedures, eliminate redundant testing and eradicate multiple testing programs. This is as relevant today as it was then.

The **Unified Testing Criteria** is a 'living and breathing' document, which provides the benchmark against which apps should be tested. Without the Java Verified approval, there's no guarantee that the app won't work. More explicitly, the Java Verified endorsement shows the world that your app **does** work.

Over the past couple of years – partly as a result of the influx of Application Stores / Shops, and the result deluge of apps into the market - the need for apps to reach a certain, industry-accepted level of quality has increased dramatically. And this applies to **all** apps, irrespective of their simplicity or complexity.

The Java Verified accreditation means (to developers, 'retailers' and consumers alike) that the apps concerned work, and meets or surpasses of a level of quality considered to be the industry standard.

Would you buy a product from an established retailer if you thought there was a risk that it hadn't gone through rigorous QA? Neither would we!

Q. But how much will it cost me?

A. First of all you'll need to buy a **Publisher ID** from TC TrustCenter (the people who run the Submission Portal) and pay a test house to test your app(s).

The cost for this is 200 USD and the Publisher ID lasts a year.

If you want to test your application as if they were signed, use an **R&D Signature**, which can be obtained through the Submission Portal when you have the Publisher ID.

There is no cost for this.

When you are sure that your application complies with the test criteria you can send it off to the test house via the Submission Portal.

The cost for this varies depending on which test house you use, the complexity of your app, and whether you're submitting it for testing or re-testing.

The Java Verified signing is included in the cost of the testing and done after your application passes the testing.

Therefore, there is no additional cost for this.

Everything you pay is paid direct to the vendors involved.

Java Verified takes none of the money.

Q. I'm a small developer with a few, simple apps: for example, 'Sam the Squeaker.' You just press a '1' on your device, and Sam squeaks. That's it: great fun, but very simple! It'll cost me a fortune to go through all the JV testing.

A. It won't.

We're aware that the industry has changed dramatically in the last 18 months, resulting in a large number of smaller developers developing very simple apps:

- simple apps requiring no permissions.
- stub apps which just open internet links
- simple apps with ads
- demo games / apps with wap site access to facilitate the downloading of the full game.

It's clear that the full set of tests that complex apps need to go through isn't applicable to these simple apps.

Therefore, we've recently introduced a smaller set of tests, suitable for a simple app. Of course, you still have to go through a test house to get your apps tested, but the cost of doing this will be a lot lower than for a more complex app.

And of course, what we at Java Verified have done is to get an approximate cost from some test houses. It's up to you to negotiate with them, based on the number of simple apps you're submitting, and the complexity of those apps.

Q. OK, but I'll still need to send every version of every app for every device through to be tested won't I?

A. We're working on this, but meanwhile, you need to look at reducing the number of app versions you produce. We can help you with this. Take a look at our defragmentation guidelines. Also, in 2009 we amended the Unified Testing Criteria (featuring the criteria the test houses use to test your app) to reduce the number of devices your apps need to be tested against. This reduced the cost for you, and we're working on reducing it further.

Q. I'm a large developer, developing and commercialising hundreds of app titles each month. I work directly with the operators and device manufacturers who sell my apps. I ingest my apps direct to them. However, I know I still have to get them JV signed before they'll accept them. Why should I pay the test house for each app I want testing? Surely you trust me to do this myself?

A. Yes, we do trust you! On August 23 2010 we announced Trusted Status for developers: a privileged status to be granted to developers who will have proven that the quality of their Java ME apps is of a consistently high standard. These are developers who will have earned the trust of Java Verified by demonstrating unflinchingly that testing to the UTC standard is a crucial part of their product development activity.

Java Verified developers who are awarded Trusted Status **no longer** need to use one of the three accredited test houses to test their applications, and **no longer** need to declare the test results. They have gained the trust of Java Verified that their apps will consistently pass the criteria.

They upload their apps into the Submission Portal as normal, but instead of sending them away to a test house before we sign your apps with the JV signature, they skip the test house part because we know they'll have already tested their apps to the JV standard.

Trusted Status developers submit them to us through a version of the Submission Portal, and we sign the apps for them straight away.

Java Verified conduct random audits on the apps to verify that the developers are meeting the level of quality that had been pre-agreed with Java Verified. If the apps fail the audit, we remove the developer's trusted status. Fair enough?

This is all about rewarding developers who constantly meet the high testing standards of Java Verified.

For full information about Trusted Status, look in the Docs & FAQs section of the website.

Q. If I'm an operator or device manufacturer selling apps, I guess know I should be insisting that each app / game I accept is JV approved, but developers keep telling me how expensive it is. I can't be bothered to insist on it.

A. First of all, because of the introduction of simple app testing and Trusted Status, it isn't expensive. Let us explain:

If you're ingesting apps from individual developers, they can now test their apps through the 'simple app testing' scheme described above. It'll cost them a lot less than it currently does.

If you're ingesting a load of apps through an aggregator, the aggregator may be granted Trusted Status by Java Verified. If you'd like them to gain the status, tell us about them.

And if you're getting your apps from one of the large publishing houses (who are producing their own apps and games), then there's a high chance they'll soon be granted Trusted Status by Java Verified anyway. Check with us that this is the case. Again, if you can vouch for them, we'll take a close look at their testing record, and consider letting them have it.

Q. OK, so that makes it easier and cheaper, but remind me – why should I insist on getting my developers' apps Java Verified?

A. It's in everyone's interest to let consumers have the best in terms of app quality. Java Verified has developed the criteria against which apps should be tested, and the programme rewards apps that meet this criteria, by allowing them to use the **Java Powered™** logo. This shows the consumer that the quality standard has been met.

Not only does this put everyone involved in a good light, but – because the apps will work - it also means that your customer helpline won't be blocked up with irate consumers complaining that 'your' apps don't work.

And finally, if you bring in apps that have been approved to Java Verified standard, it means that the amount of testing you need to do yourself is reduced.

How can all of that not be a good thing?

Q. Fine for the consumer, but how do I (the operator / device manufacturer) know that the JV standard has been achieved? Developers *don't have to* use the Java Powered™ logo, and the app could surely have been signed by another CA, and not through JV.

A. If the app has successfully been through the Java Verified programme, the JAD file will have been amended to include reference to this. That's how you'll know. And once the application itself has been signed, this certifies that it has been tested, and the level of quality achieved. You can then be assured that the application has not been changed since being tested.

Q. Tell me more about the Java Powered™ logo. Who can use it, and under what conditions?

A. The use of the **Java Powered™** logo shows the world that an app is Java Verified. If the developer wants to integrate the Java Powered™ logo into their app, they'll need to have embedded it into the app prior to submitting it into the Submission Portal.

But that doesn't necessarily mean they'll be able to use it. Let us explain ...

If the app passes the tests outlined in the Java Verified **Unified Testing Criteria**, the app will be signed. If the developer has installed the Java Powered™ logo into the app, then all is well – they'll be able to use it. Whether they use the logo or not, the JAD file will need to be amended (before the app is signed) to reflect the app's Java Verified status.

If, however, during the app testing process, the app doesn't pass the section that checks the application's likeliness of being found unfit for publication based on local laws, then they won't be granted permission to use the Java Powered™ logo, and will be asked to remove it before their app is signed. Assuming all other tests are passed however, the app's Java Verified status will still be reflected in the JAD file.

Q. So, can I add the Java Powered™ Logo to my app after my app has passed the tests?

A. No. It has to be done prior to submitting it.

Q. Assuming I get permission to use it, can I also use it in my marketing materials?

A. Absolutely. And we encourage you to shout about it as much as you can. It's your 'badge of honour'.

There are many pdfs and videos on the Java Verified website giving more information about the terms that were highlighted in this document in **bold**.

Either take a look in the **Docs, FAQs and other info** section to find them all, or just link to them from the **Relevant Resources** sections throughout the site.