

LatencyCheck

Stegvis funktionsbeskrivning

Allmänt flöde för installation, start, kontroll, trace, analys, resultat, uppdatering och avinstallation

Detta dokument beskriver steg för steg hur LatencyCheck fungerar från installation till daglig användning och avinstallation. Beskrivningen är generell och beror inte på en viss build-version.

Grundprincip för bedömning: programmet "straffar" inte systemaktivitet. Det kontrollerar om aktiviteten påverkar funktionell respons, latency, queue/backlog, DPC/ISR-fynd och praktiska symtom.

1. Översikt över funktionen

Fas	Vad händer
1	Användaren installerar programmet med ett setup som förbereder behörigheter, filer, genvägar och hjälpkomponenter.
2	Programmet startar som huvud-GUI och kontrollerar att villkoren för latency diagnostics finns.
3	Användaren väljer den kontroll eller trace som ska köras.
4	Programmet samlar data, visar framsteg och skapar ETL-filer och/eller rapporter.
5	Data analyseras och omvandlas till läsbara resultat.
6	Bedömningen klassificerar fynd som OK, aktivitet, warning eller alert efter påverkan på responsen.
7	Aktuella resultat visas i Results/Summary och senaste resultat sparas i History.
8	Användaren kan söka uppdateringar, byta språk eller avinstallera programmet.

2. Installation och förberedelse

Steg	Beskrivning
2.1	Användaren kör programmets setup.
2.2	Setup kontrollerar administrator/UAC-rättigheter där det krävs.
2.3	Språk och grundläggande installationsval väljs.
2.4	Krav som runtime, trace-verktyg och systemrättigheter kontrolleras.
2.5	Om äldre installationsrester finns rensas de före installationen.
2.6	Körbara filer, ikoner, hjälpskript och nödvändiga resources kopieras till installationsmappen.

2.7	Start Menu/Desktop-genvägar, App Paths, uninstall entries och vid behov en elevated scheduled task skapas.
2.8	Efter slutförande kan programmet öppnas direkt eller via genväg.

3. Programstart

Steg	Beskrivning
3.1	Användaren öppnar LatencyCheck från Start Menu, Desktop-genväg eller fastnålad taskbar-ikon.
3.2	Programmet kontrollerar att det körs med rätt rättigheter för diagnostics.
3.3	Om elevated körning krävs startas det om via den avsedda mekanismen så att GUI öppnas korrekt.
3.4	Huvudfönstret laddas och flikar, knappar, sökvägar och språkval initieras.
3.5	Programmet visar aktuellt läge: krav, tillgängliga verktyg, senaste rapporter och klara åtgärder.

4. Kontroll av krav och systemläge

Innan en verklig trace startar kontrollerar programmet att systemet kan ge tillförlitliga data. Det hindrar felaktiga slutsatser från en ofullständig miljö.

Kontroll	Syfte
Rättigheter	Bekräfta rättigheter för trace-insamling och åtkomst till systemdata.
WPT / Xperf	Avgöra om verktygen för trace och DPC/ISR-analys finns.
Runtime	Bekräfta att nödvändig .NET/Desktop runtime finns eller kan installeras manuellt.
Mappar	Säkerställa mappar för Traces, Results, History och hjälpfiler.
Tidigare resultat	Ladda eller rensa aktuella visningar så gamla resultat inte blandas med nya.

5. Trace / mätning

Steg	Beskrivning
5.1	Användaren väljer trace-åtgärden i GUI.
5.2	Programmet förbereder lagringsmappen och anger ETL-/rapportnamn med timestamp.
5.3	Datainsamling startar och framsteg visas.
5.4	Under mätningen undviks för tidig bedömning enbart på grund av activity.

5.5	Mätningen stoppas när tid/åtgärd är klar eller när användaren stoppar den.
5.6	ETL och tillhörande rapporter skrivs till avsedda mappar.
5.7	Programmet går från insamling till analys.

6. Dataanalys

Analysen gör rå trace-data till läsbara slutsatser. Det viktiga är inte bara vilken driver som varit aktiv, utan om aktiviteten hänger samman med verklig fördröjning eller felaktig respons.

Steg	Vad kontrolleras
DPC / ISR	Tider, drivers/modules, koncentrationer, toppar och möjliga fördröjningspunkter.
ETL trace	Samband mellan aktivitet, tidsbeteende och intressanta punkter.
Drivers / modules	Moduler med systematisk eller ovanlig belastning identifieras.
Focus modules	De viktigaste fynden isoleras så användaren inte går vilse i ovidkommande information.
Slutsats	Tekniska mätningar görs om till praktisk indikation OK, uppmärksamhet eller problem.

7. Bedömningslogik

Programmet använder funktionell bedömning. Indikationen bygger inte bara på att något rör sig i systemet, utan på om rörelsen skapar ett responsproblem.

Latency Check. Mätgränserna ändras inte; endast texten blir mer precis så att en tillfällig topp inte förväxlas med aktiv belastning.

Status	Betydelse	Var det visas
OK	Ren mätning / operativ latens ej bekräftad.	Kontroll, jämförelse, Rekommendationer och rapporter.
OK - AKTIV AKTIVITET	Aktivitet finns, men utan dokumenterad effekt på funktionell respons.	Kontroll och Rekommendationer.
VARNING - TOPP	Det aktuella värdet är tillbaka till OK, men högsta/toppvärdet passerade gränsen under mätningen.	Kontrollstatus, gammal/ny status i jämförelse och rekommendationsorsak.
VARNING - KORRELATION	Varningen kommer från en kombination av indikatorer, som CPU-kö, DPC/IRQ, diskfördröjning eller nätverks-backlog.	Rekommendationer och sammanfattande slutsatser.
AKTIV VARNING	Det aktuella värdet ligger fortfarande över gränsen. Belastningen finns nu.	Kontroll, jämförelse, Rekommendationer och rapporter.
AKTIVT LARM	Starkt eller bekräftat responstryck i detta tidsfönster.	Kontroll, jämförelse, Rekommendationer och rapporter.

Effekt på Jämförelse och Rekommendationer

- Fliken Rekommendationer använder samma klassificering i totalstatus, betydelse, förslag, korrelation med Kontroll och orsak.
- Fliken Jämförelse använder den för gammal och ny mätstatus. Numeriska differensfält förblir numeriska.
- Ändringen påverkar endast tolkningstexten. Den ändrar inte gränser, färger, räknare, trace-logik, Worker-logik eller installationsflöde.

8. Resultatvisning

Område	Roll
Summary	Visar huvudslutsatserna från aktuell mätning samlat.
Results	Behåller resultatet från senaste aktiva analys.
History	Sparar senare mätningar för jämförelse med tidigare resultat.
Reports	Öppnar eller visar producerade rapportfiler.
Recommendations	Ger praktiska råd när ett fynd kräver åtgärd.
Drivers / IRQ / ISR	Visar specialfynd för drivers, interrupts och modules.

9. Språkbyte och UI-uppdatering

Steg	Beskrivning
9.1	Användaren byter språk i UI.
9.2	Programmet tillämpar översättningar på huvudfönster och flikar.
9.3	Fönster eller hjälpvisningar med gammal text stängs/rensas så blandade språk inte finns kvar.
9.4	Användaren öppnar enskilda resultat/verktyg igen på det nya språket.

10. Programuppdatering

Steg	Beskrivning
10.1	Användaren väljer kontroll av ny version.
10.2	Programmet kontrollerar uppdateringskällan och jämför lokal version med tillgänglig.
10.3	Om en nyare version finns informeras användaren och nedladdning/installation visas.
10.4	Nedladdning/uppdatering sker i kontrollerat flöde och framsteg visas vid behov.

10.5	Efter uppdatering måste den nya versionen synas korrekt i UI, About, update check och installer metadata.
-------------	---

11. Avinstallation

Steg	Beskrivning
11.1	Användaren startar uninstall från Start Menu, Windows Apps/Programs eller relevant uninstall entry.
11.2	Avinstalleraren begär rättigheter där det behövs.
11.3	Den stänger programmet om det körs.
11.4	Den tar bort scheduled task, genvägar, App Paths och registry uninstall entries.
11.5	Den rensar installationsmappen säkert.
11.6	Där det är avsett kan results/history bevaras eller flyttas i stället för att förloras direkt.

12. Fullt stegflöde från användare till resultat

#	Flöde
1	Installera LatencyCheck.
2	Öppna programmet från genväg eller Start Menu.
3	Kontrollera krav och rättigheter.
4	Välj språk och bekräfta UI.
5	Välj kontroll-/trace-typ.
6	Starta mätning med framsteg.
7	Samla ETL/DPC/ISR-data.
8	Analysera drivers/modules/tider.
9	Skapa rapporter och sammanfattning.
10	Visa resultat med funktionell bedömning.
11	Öppna enskilda reports eller recommendations.
12	Spara aktuellt resultat och uppdatera history.
13	Kontrollera eventuellt uppdatering.
14	Avinstallera eventuellt när det inte längre behövs.

13. Kärnidé

LatencyCheck fungerar som en guide för responsdiagnos: det organiserar datainsamling, döljer komplexiteten i trace-verktyg, visar viktiga fynd och undviker att kalla normal systemaktivitet för problem.

Målet är inte bara att se vilken driver som rörde sig mest, utan att visa om verklig fördröjning påverkar datoranvändningen.

Dessutom visar den fullständiga DPC/ISR-drivrutinslistan drivers/modules från den inlästa DPC/ISR trace med ISR/DPC-antal, ISR/DPC CPU-tid, total CPU-tid och företag/produkt, så att användaren ser vilka drivrutiner som bidrog mest i mätningen utan att ändra utvärderingslogiken.