

LatencyCheck

Поетапно описание на работата

Общ поток на инсталация, стартиране, проверка, trace, анализ, резултати, актуализация и деинсталация

Този документ описва как работи LatencyCheck от инсталацията до ежедневната употреба и деинсталацията. Описанието е общо и не зависи от конкретна build версия.

Основен принцип: приложението не наказва самата активност на системата. То проверява дали активността влияе върху функционалния отговор, латентността, queue/backlog, DPC/ISR находките и практическите симптоми.

1. Общ преглед

Фаза	Какво се случва
1	Потребителят инсталира приложението със setup, който подготвя права, файлове, shortcuts и помощни компоненти.
2	Приложението стартира като основен GUI и проверява условията за latency diagnostics.
3	Потребителят избира проверката или trace, който да се изпълни.
4	Приложението събира данни, показва прогрес и създава ETL файлове и/или отчети.
5	Данните се анализират и се превръщат в четими резултати.
6	Оценката класифицира находките като ОК, активност, warning или alert според ефекта върху отговора.
7	Текущите резултати са в Results/Summary, а последните се пазят в History.
8	Потребителят може да провери за актуализации, да смени език или да деинсталира приложението.

2. Инсталация и подготовка

Стъпка	Описание
2.1	Потребителят стартира setup на приложението.
2.2	Setup проверява administrator/UAC права, когато са нужни.
2.3	Избират се език и основни опции за инсталация.
2.4	Проверяват се runtime, trace инструменти и системни права.

2.5	Ако има стари остатъци от инсталация, те се почистват предварително.
2.6	Изпълними файлове, икони, помощни скриптове и resources се копират в инсталационната папка.
2.7	Създават се Start Menu/Desktop shortcuts, App Paths, uninstall entries и при нужда elevated scheduled task.
2.8	След края приложението може да се отвори веднага или чрез shortcut.

3. Стартиране на приложението

Стъпка	Описание
3.1	Потребителят отваря LatencyCheck от Start Menu, Desktop shortcut или закрепена taskbar икона.
3.2	Програмата проверява правилните права за diagnostics.
3.3	Ако е нужно elevated изпълнение, се прави relaunch чрез предвидения механизъм.
3.4	Зарежда се основният прозорец и се инициализират табове, бутони, пътища и език.
3.5	Приложението показва състояние: изисквания, налични инструменти, последни отчети и готови действия.

4. Проверка на изисквания и системно състояние

Преди истински trace приложението проверява дали системата може да даде надеждни данни, за да се избегнат грешни изводи от непълна среда.

Проверка	Цел
Права	Потвърждение на права за trace и достъп до system data.
WPT / Xperf	Проверка за налични инструменти за trace и DPC/ISR анализ.
Runtime	Потвърждение за необходимия .NET/Desktop runtime или ръчна инсталация.
Папки	Осигуряване на папки Traces, Results, History и помощни файлове.
Предишни резултати	Зареждане или почистване на изгледите, за да не се смесват стари и нови резултати.

5. Изпълнение на trace / измерване

Стъпка	Описание
5.1	Потребителят избира trace действие в GUI.

5.2	Приложението подготвя папка и име на ETL/отчет с timestamp.
5.3	Събирането на данни започва и прогресът се вижда.
5.4	По време на измерването не се прави прибързана оценка само заради activity.
5.5	Измерването спира след време/действие или при спиране от потребителя.
5.6	ETL и отчетите се записват в предвидените папки.
5.7	Приложението преминава от събиране към анализ.

6. Анализ на данни

Анализът превръща raw trace данните в разбираеми заключения. Важно е не само кой driver е активен, а дали активността е свързана с реално забавяне или лош отговор.

Етап	Какво се проверява
DPC / ISR	Времена, drivers/modules, концентрации, пикове и възможни точки на забавяне.
ETL trace	Връзка на активността с времево поведение и важни точки.
Drivers / modules	Модули със систематично или необичайно натоварване.
Focus modules	Отделяне на най-важните находки от несвързана информация.
Заключение	Превод на техническите измервания в практично ОК, внимание или проблем.

7. Логика на оценка

Приложението използва функционална оценка: не гледа само движение в системата, а дали то създава проблем с отговора.

Latency Check. Праговете на измерване не се променят; само текстът става по-точен, за да не се бърка временен пик с активен натиск.

Състояние	Значение	Къде се показва
ОК	Чисто измерване / оперативна латентност не е потвърдена.	Проверка, Сравнение, Препоръки и отчети.
ОК - АКТИВНА ДЕЙНОСТ	Има дейност, но без документиран ефект върху функционалната реакция.	Проверка и Препоръки.
ПРЕДУПРЕЖДЕНИЕ - ПИК	Текущата стойност се е върнала към ОК, но най-високата/пиковата стойност е преминала лимита по време на измерването.	Състояние в Проверка, старо/ново състояние в Сравнение и причина за препоръка.
ПРЕДУПРЕЖДЕНИЕ - КОРЕЛАЦИЯ	Предупреждението идва от комбинация от индикатори, като CPU опашка, DPC/IRQ, забавяне на диска или мрежов backlog.	Препоръки и обобщени заключения.
АКТИВНО ПРЕДУПРЕЖДЕНИЕ	Текущата стойност все още е над лимита. Натискът е налице сега.	Проверка, Сравнение, Препоръки и отчети.

АКТИВНА АЛАРМА	Силен или потвърден натиск върху реакцията в този прозорец.	Проверка, Сравнение, Препоръки и отчети.
-----------------------	---	--

Ефект върху Сравнение и Препоръки

- Разделът Препоръки използва същата класификация в общо състояние, значение, предложение, корелация с Проверка и причина.
- Разделът Сравнение я използва за старото и новото състояние на измерването. Числовите полета за разлика остават числови.
- Промяната засяга само текста за тълкуване. Не променя прагове, цветове, броячи, трасе логика, Worker логика или инсталационен поток.

8. Показване на резултатите

Област	Роля
Summary	Показва основните изводи от текущото измерване.
Results	Пази текущия резултат от последния активен анализ.
History	Пази последните измервания за сравнение.
Reports	Отваря или показва създадените отчети.
Recommendations	Дава практически указания при нужда от действие.
Drivers / IRQ / ISR	Показва специализирани находки за drivers, interrupts и modules.

9. Смяна на език и обновяване на UI

Стъпка	Описание
9.1	Потребителят сменя езика от UI.
9.2	Приложението прилага преводи в основните прозорци и табове.
9.3	Старите текстове в помощни изгледи се затварят/чистят, за да няма смесени езици.
9.4	Потребителят отваря резултати или инструменти отново на новия език.

10. Актуализация на приложението

Стъпка	Описание
10.1	Потребителят избира проверка за нова версия.
10.2	Приложението проверява източника за актуализация и сравнява локалната версия с наличната.
10.3	При по-нова версия показва опция за download/инсталация.
10.4	Download/актуализацията протича контролирано с прогрес при нужда.

10.5	Новата версия трябва да се вижда правилно в UI, About, update check и installer metadata.
------	---

11. Деинсталация

Стъпка	Описание
11.1	Потребителят стартира uninstall от Start Menu, Windows Apps/Programs или съответния entry.
11.2	Деинсталаторът иска права, когато е необходимо.
11.3	Затваря приложението, ако работи.
11.4	Премахва scheduled task, shortcuts, App Paths и registry uninstall entries.
11.5	Почиства инсталационната папка безопасно.
11.6	Където е предвидено, може да запази или прехвърли results/history.

12. Пълен поток от потребител до резултат

#	Поток
1	Инсталиране на LatencyCheck.
2	Отваряне от shortcut или Start Menu.
3	Проверка на изисквания и права.
4	Избор на език и потвърждение на UI.
5	Избор на тип проверка/trace.
6	Стартиране на измерване с прогрес.
7	Събиране на ETL/DPC/ISR данни.
8	Анализ на drivers/modules/времена.
9	Създаване на отчети и Summary.
10	Показване на резултат с функционална оценка.
11	Отваряне на reports или recommendations.
12	Запис на резултата и обновяване на History.
13	По избор проверка за актуализация.
14	По избор деинсталация.

13. Централна идея

LatencyCheck е водач за диагностика на отговора: организира събирането на данни, скрива сложността на trace инструментите, показва важните находки и не нарича нормалната активност на системата проблем.

Целта не е само да се види кой driver се е движил най-много, а дали реално забавяне влияе върху употребата на компютъра.

Допълнително, пълният списък с DPC/ISR драйвери показва подробно драйверите/modules от заредения DPC/ISR trace с брой ISR/DPC, CPU време за ISR/DPC, общо CPU време и фирма/продукт, за да се вижда кои драйвери участват най-много в измерването, без да се променя логиката на оценяване.