



## **Paketeringspolicy eKlient v2.9**



Version	Datum	Ändrad av	Kommentar
1.0	2013-02-18	Mattias Jacobsson	Policy beslutad
2.0	2013-10-11	Mathias Andersson	Förändring av paketnamn och numrering av internversion samt placering av denna information i MSI. Namnstandard för App-V och MSP framtagen. REBOOT, ROOTDRIVE tillagda, och även HKCU-nycklar för filreparation. Utökning av kapitlet om tester, ICE-validering har adderats. Appendix A och B har adderats.
2.1	2013-11-08	Mathias Andersson	Korrigerat fel i svensk språkkod, från SE till SV enligt ISO 639-1. Lagt med språkkoden MUL för MultiLanguage, finns inte med i ISO 639-1.
2.2	2013-11-13	Mathias Andersson	Uppdaterat avsnitt 5.3 med förhållningssätt gällande språkkoder utifrån referensgruppsbeslut 2013-11-13.
2.2	2013-11-21	Mathias Andersson	Version 2.2 godkänd av styrgruppen i LG01
2.3	2014-10-24	Mathias Andersson	Strukit referenser till eKlients changeprocess i avsnitt 2, då paketeringskraven ifrån den processen införlivats som egna punkter enligt kompletteringar nedan. Kompletterat avsnitt 3 med förhållningssätt för eKlients paketrepository. Kompletterat avsnitt 4 med förhållningssätt för eKlients paketedokumentationssite, ePaket. Kompletterat avsnitt 7 med förhållningssätt för dokumentation av tekniska tester och UAT i eKlients paketedokumentationssite, ePaket.
2.4	2015-11-26	Martin Rönnberg Pär Hagman	Lagt till kvalitetskrav för applikationer och MSI-filer samt grundregel för validering under kapitel 1. Lagt till regler för levererade MSI-filer under avsnitt 4.2 Lagt till Windows 10 specifika krav under avsnitt 4.2. Redigerat information för MSP-filer under avsnitt 4.3.2 Lagt till regel för QA under kapitel 6 Tagit bort Valideringsinformation under Appendix. Denna information kan komma



			flyttas till annat dokument i Leveransmodell LITA Justerat innehållet språkligt Justerat & kompletterat landstingskoder
2.5	2015-12-07	Pär Hagman	Justerat efter kommentarer från referensgrupp och beredningsgrupp
2.6	2016-02-14	Nicke Källén	Omorganisering för MSI / App-V. Grundläggande regler för App-V
2.6	2016-04-05	Åse Berneblad	Beslutad av Beredningsgruppen
2.6	2016-11-04	Jan Hesselgren	Lagt till att program inte skall autostarta
2.61	2017-02-21	Jan Hesselgren	Justerat kring drivrutinsignering, eklient håller inte längre med certifikat. Egen signering avrådes.
2.7	2018-02-28	Jörgen Nilsson	Tagit bort kraven på Signering av MSI. Ändrat till att MSI eller App-V är de primära formaten. Ändrat validering och hantering av ICE warnings och errors i 1.3.1.2. Ändrat alla landsting/regioner till medlemmar eller organisationer Ändrat så att 7.2 medlemskoder pekar på eKlient Portalen istället för att finnas i dokumentet. Lagt till ARPCOMMENTS i 4.2.1 från 4.2.3 för förtydligande
2.8	2019-02-12	Jörgen Nilsson	Lagt till MSIX som paketeringsformat under alla stycken det är applicerbart Ersatt alla referenser till eklient centrala repository med ePaket
2.9	2020-04-29	Jörgen Nilsson	Lagt till ikon som rekommenderad i paketeringsmedia Tagit bort referenser till centrala repository Ändrat att applikationer inte "bör" autostarta istället för "skall"



<b>1. Regelverk för konfiguration av installationsmedia</b>	<b>6</b>
<b>2. Huvudregler för linjeaktiviteten Paketering</b>	<b>6</b>
2.1 Beslutande part för val av installationsmetod	6
2.2 Allmänna regler för skrivbordsapplikationer	6
2.3 Allmänna Regler för Paket	7
2.3.1 Allmänna Regler för MSI paket	7
2.3.2 Allmänna Regler för App-V Paket	7
2.3.3 Allmänna Regler för MSIX Paket	8
2.4 Avgränsningar	9
2.5 Installationsformat	9
2.6 Funktionsmodifiering	9
2.7 SnapshotPC	10
2.7.1 App-V specifika delar	10
2.7.2 MSIX specifika delar	10
<b>3. Data Media Library (DML)</b>	<b>11</b>
3.1 Regler för DML	11
3.2 Placering av DML	11
3.2.1 eKlients paketrepository	11
3.2.2 Mappstruktur	11
<b>4. Dokumentation</b>	<b>12</b>
4.1 Regler för dokumentation	12
4.2 Paketdokumentationssite	12
<b>5. Regler</b>	<b>13</b>
5.1 Teckenförklaring	13
5.2 Regelverk per format	13
5.2.1 MSI specifika regler	13
5.2.2 App-V specifika regler	15
5.2.3 MSIX specifika regler	17
5.2.4 Paketnamn	18
5.3 Genvägar	20
<b>6. Hantering av manuella installationer</b>	<b>21</b>
6.1 Beskrivning	21
6.2 Orsak till manuella installationer	21



6.3	Regelverk för manuella installationer .....	21
6.4	Manuellt Media .....	21
<b>7.</b>	<b>Tester .....</b>	<b>22</b>
7.1	Tekniskt test .....	22
7.2	Kvalitetssäkring (QA).....	23
<b>8.</b>	<b>Bilagor .....</b>	<b>24</b>
8.1	Språkkoder .....	24
8.2	Medlemskoder .....	24
<b>9.</b>	<b>Appendix A.....</b>	<b>25</b>
9.1	Programvara .....	25



## 1. Regelverk för konfiguration av installationsmedia

Detta dokument tar upp det regelverk som ska användas vid paketering av applikationer till eKlient.

## 2. Huvudregler för linjeaktiviteten Paketering

Alla applikationer som ska installeras på en eKlient maskin ska om möjligt MSI-paketeras, sekvenseras i App-V eller paketeras som MSIX.

### 2.1 Beslutande part för val av installationsmetod

All media samt konfigurationsunderlag ska inkomma till paketerarna för behandling innan paketering får ske. Efter genomgång av media är det enbart paketerarna (i eventuellt samråd med leverantör) som tar beslut om vilken installationsmetod som ska användas för aktuell applikation.

### 2.2 Allmänna regler för skrivbordsapplikationer

Levererade applikationer skall vara utvecklade enligt Microsofts riktlinjer för skrivbordsapplikationer:

- Applikationer utvecklade för Windows 7 ska följa de rekommendationer som Microsoft specificerar i "Windows 7 software logo program" (<https://docs.microsoft.com/en-us/windows/desktop/win7appqual/certification-for-windows-7-and-windows-server-2008-r2-applications>)
- Applikationer utvecklade för Windows 8.1 och/eller Windows 10 ska följa de rekommendationer som Microsoft specificerar i "Certification requirements for Windows desktop apps" (<https://msdn.microsoft.com/library/windows/desktop/hh749939>)

Ovan nämnda regler gäller för alla applikationer, både de som är framtagna av interna resurser och de som är levererade av tredje part.



## 2.3 Allmänna Regler för Paket

### 2.3.1 Allmänna Regler för MSI paket

#### 2.3.1.1 Allmänna Regler för MSI installationer

- Alla paket ska följa "Windows Installer Best Practices" [https://msdn.microsoft.com/en-us/library/bb204770\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/bb204770(v=vs.85).aspx)

Ovan nämnda regler gäller för alla installationspaket, både de som är framtagna av interna resurser och de som är levererade av tredje part.

#### 2.3.1.2 Validering av MSI

Alla internt framtagna MSI installationer bör valideras. Validering (Full MSI Validation Suite) sker via det valda paketeringsverktygets inbyggda valideringsfunktion, alternativt via valideringsfunktionen MSIVAL2 som ingår i "Microsoft Windows SDK for Windows 7 and .NET Framework 4".

Följande regler för validering gäller för eKlient:

- Internt framtagna MSI installationer kan innehålla s.k. ICE WARNINGS\* och ICE ERRORS\* ifall dessa kan motiveras.
- Levererade MSI installationer valideras inte eftersom vi inte kan påverka dem i de flesta fallen utan de testas genom att man testar installation och avinstallation och utvärderar resultatet.

(\* ICE = Internal Consistency Evaluators)

### 2.3.2 Allmänna Regler för App-V Paket

#### 2.3.2.1 Allmänna Regler för App-V installationer

Alla App-V paket skall skapas med Microsoft Application Virtualization Sequencer.

Tredje-parts produkter kan användas för att efter-redigera paket, men grundläggande paket skall skapas med Microsoft Application Virtualization Sequencer.

Paketering bör ske på samma arkitektur som är paketet är avsett att användas på. Det är tekniskt möjligt att göra paket på 32-bitars operativsystem och därefter installera paketet på ett 64-bitars operativsystem, men det avrådes då flertalet möjligheter till integration med operativsystemet kommer att stängas av.



### **2.3.2.2 Validering av App-V Paket**

App-V paket skall gå att installera i både system-context samt användar-context

## **2.3.3 Allmänna Regler för MSIX Paket**

### **2.3.3.1 Allmänna Regler för MSIX installationer**

MSIX är en vidareutveckling av APPX formatet och samma regelverk som gäller för MSIX skall gälla för APPX. APPX kommer att försvinna helt på sikt och ersättas av MSIX fullt ut. Därav har vi valt förkortningen "X" i namnet på applikationen vilket är giltigt för både APPX och MSIX.

Alla MSIX paket måste signeras, det går inte att installera dem osignerade. När det gäller MSIX paket från leverantör skall dessa vara signerade med ett publikt Code-Signing certifikat alternativt av Microsoft.

MSIX paket som skapas internt skall signeras med antingen ett publikt Code-Signing certifikat eller ett internt Code-Signing certifikat. I det sistnämnda fallet kan den som laddar ned paketet via ePaket signera om det med sitt egna certifikat väldigt enkelt.

### **2.3.3.2 Validering av MSIX Paket**

MSIX paketet skall gå att installera i både system-context samt användar-context





## 2.4 Avgränsningar

Följande ska inte paketeras:

- Media som enbart består av genvägar till webbapplikationer, webbsidor, dokument, mappar och/eller share. Förslaget är att istället nyttja intranät, uppmappningar och liknande då detta är ett enklare och effektivare tillvägagångssätt för publicering.
- Mjukvara vars enda syfte är att kunna inhämta data ifrån kringutrustning med uttagbart minneskort (t.ex. digitalkameror).

## 2.5 Installationsformat

Följande format kan användas för central distribution:

- MSI/MST
- Legacy (Setup.exe)
- Script
- App-V 5.1+
- MSIX

MSI ska vara den teknik vi främst använder. Undantag ifrån detta kan vara när vi har applikationer som har ett annat fungerande sätt att installeras på. Exempel på detta är Office som har en egen installations metod och kan hantera växlar till sin EXE installation och köras "silent". Bundlade tredjeparts installationer, dvs. när ett paket installerar flera olika applikationer som kan installeras separat, ska inte användas annat än när det är tillrådligt, t.ex. installationssviter från leverantör.

Exempel på applikationer som ska installeras separat är .Net, Java och C++. Orsak till eventuella avsteg från denna regel ska dokumenteras i paketedokumentationen.

## 2.6 Funktionsmodifiering

Paketerare i samråd med leverantör avgör om en implementation av funktion som leverantören inte tillhandahåller som standard ska läggas till vid paketering. Tillägg av en sådan funktion får enbart göras i de fall leverantören supporterar lösningen. Detta gäller exempelvis flerdatabasstöd i en applikation som inte kan hantera mer än en databas i sitt standardutförande.



## 2.7 SnapshotPC

När ett snapshot görs (oavsett teknik) ska detta utföras på den framtagna eKlients-snapshotmallen, som har följande egenskaper:

- Maskinen har inte något annat installerat än de servicepacks/patchar för Windows som utgör lägstanivån för eKlient.
- Maskinen är inte domänansluten för att förhindra policier från att påverka

Detta är till för att undvika felaktigheter i paketen. SnapshotPC återställs till ursprungsläge efter varje gjord snapshot. I de fall där undantag görs från ovanstående, ex krav på Office-installation, domänanslutning mm, ska detta dokumenteras i paketedokumentationen.

### 2.7.1 App-V specifika delar

På en SnapshotPC skall inte Microsoft Application Virtualization Sequencer varit startad innan ett snapshot är taget. Möjligheten finns att man inte behöver installera Microsoft Application Virtualization Sequencer innan ett snapshot skapas, vilket möjliggör att byta version av paketeringsmjukvaran.

Rekommenderat är att ha en likadan nivå på snapshot av Microsoft Visual C++ Redistributables<sup>1</sup> som man har på klienterna.

Om snapshot PC måste vara medlem i domänen på grund av applikationsberoende till domänen är rekommendationen att säkerställa att Group Policy (eller skript) som pekar om delar av profilen till nätverkssökvägar undantas.

### 2.7.2 MSIX specifika delar

Microsoft tillhandahåller "MSIX Packaging tool environment" som går att ladda ned direkt i Hyper-V på en Windows 10 dator under quick create. Alternativt använder man sig egen SnapshotPC.

MSIX Packaging Tool kontrollerar själv att tjänster som Microsoft Update, Windows Search indexer med mera inte körs på SnapshotPC.

---

<sup>1</sup> <https://support.microsoft.com/en-us/kb/2977003>



## 3. Data Media Library (DML)

### 3.1 Regler för DML

Media för alla driftsatta applikationer ska finnas arkiverat hos varje organisation, i originalutförande samt i färdig form, inga projektfiler eller arbetsfiler laddas upp till centralt repository då detta är avvecklat.

### 3.2 Placering av DML

#### 3.2.1 eKlients paketrepository

eKlients paketrepository finns lokalt hos varje organisation eftersom det centrala repositoryt är avvecklat.

#### 3.2.2 Mappstruktur

Allt media placeras i följande mappstruktur med huvudmappen namngiven enligt <Paketnamn> (se punkt 4.3.1):

@ Original Media (media i originalutförande)

@ Work Files (filer som kan behövas vid återvinning av paketet)

Acceptanstest

App-V (filer specifika för App-V-tekniken)

1. Final Media (det distributionsklara media inklusive dokumentationen för paketet)

EXE (filer för legacy setup)

1. Final Media (det distributionsklara media inklusive dokumentationen för paketet)

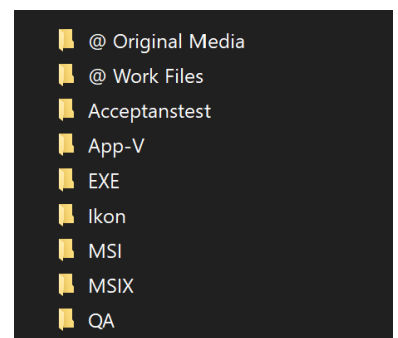
Ikon (rekommenderas)

Ikon eller ikoner om flera för applikationen i storlek 512x512, ".png" format är att föredra.

MSI (filer specifika för MSI-tekniken)

1. Final Media (det distributionsklara media inklusive dokumentationen för paketet)

MSIX (filer specifika för MSI-tekniken)





1. Final Media (det distributionsklara media inklusive dokumentationen för paketet)

QA

## 4. Dokumentation

### 4.1 Regler för dokumentation

Dokumentationen av media ska kunna användas av både paketerare och annan personal som behöver få information om paketet. Dokumentationen ska vara så pass fullständig att en ny paketering kan göras utifrån enbart dokumentationen och originalmedia.

### 4.2 Paketdokumentationssite

Dokumentation för allt media som paketeras ska göras i applikationen ePaket. Dokumentationen lagras lokalt hos respektive organisation i ePaket



## 5. Regler

### 5.1 Teckenförklaring

[ ] – MSI-Property (endast tillämpliga i MSI-paket)

< > - Programspecifika variabler

% % - Systemspecifika variabler

### 5.2 Regelverk per format

#### 5.2.1 MSI specifika regler

Förutom de generella regler som finns under Allmänna regler (se punkt 1.2) används dessa eKlient specifika regler oavsett installationsmetod där det är tillämpligt:

- **MSI-paket ifrån leverantör får inte editeras.** Anpassningar ska göras i MST, detta inkluderar all lokal konfiguration och sökvägar. (MSI-specifikt)
- **Applikationspaket som är framtagna internt** Ange i dokumentationen hur lokal konfiguration ska implementeras via transforms-fil (MST).
- All **licensinformation och/eller konfiguration** som inte är generell för alla organisationer ska läggas i en MST och [PROPERTY] ska, om möjligt, användas för att hålla denna information. Hur man använder sig av dessa Properties ska dokumenteras. I de fall det inte är möjligt att nyttja Properties men licensinformation/konfigurationens plats är känd så beskrivs detta i dokumentationen. (MSI-specifikt)
- Om egna **Custom Actions, MSI-properties eller Components** skapas ska dessa namnges med eKlient\_ följt av förklarande namn ex. eKlient\_DeleteFolder. Egna properties görs företrädesvis publika (versaler i hela property-namnet). Information om funktionen ska dokumenteras i paketedokumentationen. (MSI-specifikt)
- **Applikationen ska fungera i en nedlåst miljö.** För organisationer som använder Flexera AdminStudio ska upplåsning av specifika delar på lokal disk eller i registret ske vid MSI-paketering via inbyggd "Custom InstallShield Handling" då behörighetstabellen inte skrivs sönder vid installation och för att enkelt kunna se var rättigheter satts. Används annat paketeringsverktyg eller vid annan installationsmetod kan samma upplåsning göras med hjälp av



- Secedit. Oavsett metod som används måste upplåsningar dokumenteras i paketedokumentationen.
- **Sitelicensnycklar eller motsvarande läggs in i paketet.** Applikationer med enanvändarlicenser kan paketeras men licensen får inte bakas in i paketet. Viktigt att dokumentera vart eventuella inbakade licensnycklar finns i paketet, samt att anpassa <Paketnamn> utifrån detta.
  - Installation/avinstallation ska **ske tyst**, dvs. utan interaktion med användaren.
  - Programmen ska installeras i största möjliga utsträckning under **%ProgramFiles%** eller **%ProgramFiles(x86)%**.
  - Alla eventuella **automatiska uppdateringsfunktioner** ska **inaktiveras**.
  - **Mergemoduler** ska användas i de fall där paketeringsverktyget förslår detta och där det är lämpligt. (MSI-specifikt)
  - **Hårda sökvägar** så som "C:\Program Files" är inte tillåtna. Använd istället s.k. Formatted data type (t.ex. [#filnamn]) för att lägga till sökvägar.
  - Eventuella bundlade **tredjepartsprodukter** ska paketeras separat och inte ihop med paketet för att undvika problem med andra program. Detta gäller t ex. InstallShield Scripting Runtime och .NET Framework.
  - **[ALLUSERS]** sätts till 1, såvida annat inte krävs av programmets funktionalitet på något sätt och att det inte går att komma runt det. (MSI-specifikt)
  - **[ARPNOMODIFY]**, **[ARPNOREMOVE]** sätts till 1 (inaktiveras). **[ARPNOREPAIR]** sätts till 0 (aktiveras) om det i originalmedia inte är inaktiverat. (MSI-specifikt)
  - **[ROOTDRIVE]** sätts till C:\ (MSI-specifikt)
  - **[REBOOT]** sätts till ReallySuppress, kräver applikationen omstart vid installation/avinstallation ska detta dokumenteras i paketedokumentationen. (MSI-specifikt)
  - **[ARPCOMMENTS]** – Anges enligt <Paketnamn> (MSI-specifikt) anges i MSI eller MST om det är en leverantörs MSI. Detta fältet kan sedan plockas upp i Deploying för att säkerställa rätt namnstandard i alla system.
  - Filer tillhörande snapshotade/egenskapade MSI-paket ska inte lagras i ett öppet arkiv utan direkt i MSI eller tillhörande **Cab-filer**. (MSI-specifikt)
  - **Drivrutiner** måste signeras för att möjliggöra installation. Det är mycket viktigt att man går tillbaka till leverantören och får denna att signera drivrutinerna. Att signera drivrutiner själv är inte lämpligt. Om detta trots allt är nödvändigt görs det med certifikat som tillhandahålls av medlemmen själv. Instruktioner för hur detta görs finns på eKlient Portalen.
  - Registernycklar till användarprofilen som används som keypath för filer som ska installeras till användarprofilen via MSI-reparation skall installeras till:  
HKEY\_CURRENT\_USER\Software\eKlient\MSI  
Repair\<Paketnamn>\componentnamn=componentdestination. (MSI-specifikt)



- **Fonter** får bara installeras under %windir%\fonts, och ska läggas till med inbyggda funktioner i våra rekommenderade paketeringsverktyg.
- **Applikationer bör inte sättas att autostarta** i registret efter omboot, efter inloggning eller i användarens startup-meny.

### 5.2.2 App-V specifika regler

Eventuella bundlade **tredjepartsprodukter** ska paketeras separat och inte ihop med paketet för att undvika problem med andra program. Detta gäller t ex. InstallShield Scripting Runtime, Microsoft .NET Framework och Visual C++ Redistributable. Det gäller även **drivrutiner** eller andra OS-komponenter.

Dessa skall installeras som förkrav innan paketering påbörjas på snapshot PC och därmed också installeras som förkrav på klienter. Beroende till tredjepartsprodukter och drivrutiner skal dokumenteras tillsammans med paket-dokumentation.

Alla eventuella **automatiska uppdateringsfunktioner** ska **inaktiveras**. Uppdatering av följande fil-typer under körning av applikationer kommer att förhindras.

*.exe*  
*.dll*  
*.ocx*  
*.com*

Register-värden som skapas under **Software\Policies** (se komplett lista nedan) kommer att exkluderas ur paketet. Värdena skall endast sättas via Group Policy och rekommendationen är att identifiera den specifika inställning som sätts, och inte sätta register-värdet via Group Policy Preference. Beroende till Group Policy skall dokumenteras tillsammans med paket-dokumentationen.

*HKCU\Software\Policies*

*HKLM\Software\Policies*

*HKLM\Software\Wow6432Node\Policies*

Standard **exkludering** bör användas i den mån det kan, men även byggas ut med följande

*HKCU\Software \Microsoft\Windows NT\CurrentVersion\Windows Messaging Subsystem*

*HKCU\Software \Microsoft\Windows NT*

Exkluderingar som läggs till eller tas bort skall dokumenteras i paket-dokumentationen. Rekommenderat är att skapa ett **template (.appvt)** för att återanvändas vid framtida paketeringar alternativt ompaketering som en del av paket-filerna.



**Genvägar** skall placeras på rätt plats under installations-fasen av paketeringen för att säkerställa att framtida uppdateringar av paketet inte återgår till originalplacering av genvägen.

**Efter-redigering av paket** är inte rekommenderat att göras direkt i XML filer (DeploymentConfig, UserConfig eller Package Manifest). Rekommenderat är att säkerställa att allt konfigureras rätt under **installations-fasen** av paketeringen. Om det ej är möjligt att säkerställa att samtliga steg utförs under installations-fasen kan ändringar utföras i editerings-läge i Microsoft Application Virtualization Sequencer, alternativt att man använder tredje-partsverktyget *Gridmetric Application Virtualization Explorer*<sup>2</sup>.

Skript som skapas skall inkluderas direkt i paketet och exekvering av dem bör beskrivas i det interna manifestet i paketet. Detta är endast möjligt i App-V 5.1 och efterföljande versioner. Skript kan placeras mappen **[{AppVPackageRoot}]\Scripts** (fliken Package Files i efter-redigeringsläge).

För att definiera när skriptet skall köras behövs det **interna paket manifestet** exporteras ut ifrån fliken Advanced. Rekommenderat är att använda *Virtual Engine App-V 5 Configuration Editor* för att editera manifestet, alt *Gridmetric Application Virtualization Explorer*. När manifestet är uppdaterat kan det importeras in i paketet igen.

Följ Microsofts genomgång<sup>3</sup> för hantering av skript och dess exekvering.

Skript skall dokumenteras både vad syftet är, hur de är uppbyggda samt när de kör vid användning av paketet. Skript-filerna skall inkluderas som en arbetsfil.

De flesta mjukvaror är rekommenderade att **starta en gång** under paketeringen, specifikt under konfigurations-fasen av paketeringen. Mjukvaror (typ exempel är Java-baserad mjukvaror) som skapar filer i profilen med hårdkodade sökvägar till användar-profilen är exempel på när denna rekommendation inte bör följas.

Möjligheten att starta applikationer förekommer två gånger under paketeringen (konfiguration samt strömmning). Delen vid strömmning är valfri och bör endast användas om man inte tänker för-populera App-V cachén på klienterna. Rekommenderat vid för-cachning är att ej starta applikationen under delen för strömmning.

---

<sup>2</sup><http://www.gridmetric.com/products/ave.html>

<sup>3</sup> <https://docs.microsoft.com/en-us/microsoft-desktop-optimization-pack/appv-v5/about-app-v-51-dynamic-configuration?redirectedfrom=MSDN>





Paket skall **inte** begränsas till **specifika OS**. Paket skapade på en x64-bitars arkitektur kommer ej att installeras på en x86-arkitektur. Paket skapade på x86 kommer att installeras på en x64-arkitektur, men vissa integrationstyper kommer ej att aktiveras när paketet installeras.

### 5.2.3 MSIX specifika regler

MSIX är ett mycket mer kontrollerat format än MSI och APP-V med vad som går att göra och inte. I skrivande stunds stöds inte tjänster som exempel.

Microsofts MSIX Packaging Tool som har använts som referens hanterar inställningar i Packdatorn automatiskt som Windows Update skall vara avstängt, Windows Search Indexer med mera.

**Genvägar** skall placeras på rätt plats under installations-fasen av paketeringen för att säkerställa att framtida uppdateringar av paketet inte återgår till originalplacering av genvägen.

PackageName kan inte innehålla mellanslag vilket däremot Package Display Name kan därför är dessa samma fast Package Name inte innehåller mellanslag.

Ett MSIX Modification Package kan innehålla exempelvis en plug-in till en applikation eller annat tillägg men det kan även bestå endast av inställningar för en MSIX från leverantör.

Detta skall avspeglas i namnet på MSIX paketet som framgår under namnstandard i detta dokument.



## 5.2.4 Paketnamn

<Paketnamn> används till att namnge

- Egenskapad MSI-fil
- MST-fil
- App-V-fil
- MSIX-fil
- Huvudmappnamn på virtuell DML (se punkt [2.2.2](#))
- Paketnamn i beställnings/paketedokumentationsportal

<Paketnamn> = [ProductName] <Regional identifiering> <Lokal identifiering>  
<Språkkod> <MAN> <S> <Arkitektur> <Internversion><A> | <X> | <XM>  
[ProductVersion].

Exempel: (64-bitars version anpassad för FSS - Frölunda Specialistsjukhus i Västra Götalandsregionen för manuell installation)

Microsoft Office 2007 Professional VGR FSS SV MAN x64 R01 12.0.4518.1014

Exempel: (32-bitars version utan lokala anpassningar baserad på App-V-tekniken)

Microsoft Office 2007 Professional R01A 12.0.4518.1014

Exempel: (32-bitars version utan lokala anpassningar baserad på MSIX-tekniken)

Microsoft Office 2019 Professional R01X 12.0.4518.1014

[ARPCOMMENTS] – Anges enligt <Paketnamn> (MSI-specifikt)

[ProductName] – Befintligt värde i MSI ifrån tillverkare ändras inte.

I de fall det handlar om att applikationspaketet som har byggts via snapshot då sätts [ProductName] till det namn som visas för aktuell applikation i "Kontrollpanelen\Alla objekt på Kontrollpanelen\Program och funktioner" när applikationen är installerad manuellt

Visas det inget i "Program och Funktioner" anges [ProductName] till applikationsnamnet som vanligtvis visas i menyvalet Hjälp | Om... i applikationen. Finns inte det valet heller används "Produktnamn" i egenskaper för filen.

I de fall som ett applikationspaket som byggs från lösa resurser (filer och/eller registervärden) då sätts värdet för [ProductName] i samråd med kunden.

<Regional identifiering> - Plats för att identifiera om ett paket har anpassningar för en specifik organisation. Lämnas denna tom får inte paketet innehålla några



verksamhetsspecifika konfigurationer. Lista över organisationskoder finns på eKlient Portalen.

**<Lokal identifiering>** – Plats för att identifiera de olika varianter av ett paket som kan finnas inom respektive organisation. <Lokal identifiering> får inte förekomma om <Regional identifiering> är tom. Lämnas denna tom får inte paketet innehålla några förvaltningsspecifika konfigurationer. Detta värde ska anges vid beställning och bör vara en förkortning om max 4 tecken.

**<Språkkod>** – När en redan befintlig applikation behöver paketeras med alternativt språk ska språkkod anges enligt punkt [7.1](#) för det kompletterande paketet.

**<MAN>** – MAN anges i de paket som ska installeras manuellt. Lämnas annars tom.

**<S>** - Om ett paket är avsett för en server sätt ett S på denna plats, annars lämnas denna tom.

**<Arkitektur>** – När en 64-bitarsapplikation behöver paketeras ska x64 anges. Lämnas annars tom.

**<A>** - I de fall App-V används som teknik ska tillägget A sättas i direkt anslutning till <Internversion>, exempelvis R01A. Lämnas annars tom.

**<X>** - I de fall MSIX används som teknik ska tillägget X sättas i direkt anslutning till <Internversion>, exempelvis R01X. Lämnas annars tom.

PackageName kan inte innehålla mellanslag därför skrivs PackageDisplayName ihop under Package Name. Exempelvis:

**MicrosoftOffice2019ProfessionalR01X12.0.4518.1014**

**<XM>** - används i de fall det är ett MSIX Modification Package som endast är inställningar till ett MSIX paket från en leverantör. Lämnas annars tom.

**<Internversion>** – Den interna versionshanteringen beskriver vilken release paketet har och startar alltid med R01. Vid varje ny förändring höjs versionsnumret ett steg. Detta måste återspeglas i folder strukturen (se punkt [2.2.2](#))

**[ProductVersion]** – Befintligt värde i MSI ifrån tillverkare ändras inte. I de fall det handlar om en applikation som snapshotas alternativt Legacy (setup.exe) sätts [ProductVersion] till versionen som visas för aktuell applikation i "Kontrollpanelen\Alla objekt på Kontrollpanelen\Program och funktioner". Visas det inget i "Program och Funktioner" anges [ProductVersion] till versionen som vanligtvis visas i menyvalet Hjälp



| Om... i applikationen. Finns inte det valet heller används "Produktversion" i egenskaper för filen.

#### 5.2.4.1 MSP (MSI-specifikt)

Om det är möjligt, välj s.k. "full file" uppgraderingar före uppgraderingar i MSP-format

- En MSP-fil får inte ändra [ProductName] eftersom denna Property endast får ändras vid en s.k. major upgrade. Inträffar detta, kontakta leverantören för åtgärd.
- Appliceras en MSP-fil som ändrar [ProductVersion] ska ett nytt paket skapas (se [4.3](#) för att få ett unikt paketnamn).
- Appliceras en MSP-fil som inte ändrar [ProductVersion] ska ett nytt paket skapas och <Internversion> höjas med 1 (se punkt [4.3](#) för att få ett unikt paketnamn).

### 5.3 Genvägar

Nedanstående gäller för MSI och ska om möjligt uppfyllas med övriga installationstekniker:

- Inga startikoner/genvägar ska publiceras på skrivbordet eller i snabbstartsfältet.
- Inga avinstallationsgenvägar ska publiceras.
- Inga startikoner ska publiceras under respektive användarkonto.
- Övriga startikoner som skapas av originalmedia ligger kvar på originalplats.
- I de fall startikoner/genvägar måste skapas ska de läggas under Start-menyn\ (Alla) Program\[ProductName] för alla användare.

(Detta är en riktlinje som ska följas så långt det går. Avsteg från riktlinjen kan dock göras vid behov och dokumenteras då i paketeringsdokumentationen).



## 6. Hantering av manuella installationer

### 6.1 Beskrivning

I de fall en automatiserad installation inte är möjlig kan manuell installation användas.

Beslutet om att en applikation ska hanteras manuellt kan enbart tas av paketeringsansvariga. Nedanstående punkter gäller i dessa fall.

### 6.2 Orsak till manuella installationer

Det kan finnas ett antal orsaker till att en applikation måste installeras manuellt. Några exempel på dessa orsaker kan vara:

- Omständigheter som omöjliggör att applikationen kan distribueras centralt.
- Omständigheter som omöjliggör att applikationen kan paketeras.
- Applikationen kräver interaktion med slutanvändaren under installationsprocessen.

### 6.3 Regelverk för manuella installationer

Anpassningar i media som ska installeras med manuell metod ska göras för att i största möjliga mån överensstämmer med våra vanliga regler (se punkt 4). Tillägget MAN läggs till innan internversionen i <Paketnamn>.

Manuella installationer kopieras från den centrala distributionsplattformen till kunddatorn och genomförs därefter av supporttekniker. Installationsförfarandet ska göras enligt den installationsanvisning för den specifika applikationen som framtagits av beställare.

### 6.4 Manuellt Media

Media och lösningar för eventuella manuella installationer ska inte laddas upp på den centrala repositoryt.



## 7. Tester

### 7.1 Tekniskt test

Följande steg ska testas av paketerare för att säkerhetsställa installationen:

- Installation av applikationen i system context.
  - a. Installation av applikation för användare (App-V specifikt)
  - b. Installation av applikation för användare (MSIX specifikt)
  
- Reparation av installation (där tillämpligt).
- Start av applikation som flera nedlåsta användare (där tillämpligt).
- Påloggning av nya användare (med ej tidigare skapad profil) efter installation av paket.
- Avinstallation av applikationen i system context.
  - a. Avinstallation av applikation för användare (App-V specifikt)
  - b. Avinstallation av applikation för användare (App-V specifikt)
  
- Alla applikationer inklusive installationslogik ska testas på Windows 10 x64 oavsett vad som angetts i beställningen. Detta är för att slippa testa om alla applikationer senare. Detta resultat ska dokumenteras i applikationen ePaket.



## 7.2 Kvalitetssäkring (QA)

För att säkerställa en genomgående hög kvalitet i de gemensamma leveranserna till eKlient så skall ett färdigt installationspaket alltid testas i en produktionslik miljö och av någon annan än den som gjort paketet.

- QA utförs normalt via applikationen ePaket i ett eget processteg och med hjälp av de testprotokoll som nås via gränssnittet
- Resultatet ska dokumenteras i applikationen ePaket.

Om man inte använder applikationen ePaket så ska följande tester utföras som ett minimum vid QA:

- Alla MSI-filer valideras enligt 1.3.1.2 Validering
- Installation av applikationen i system context.
- Reparation av installation (där tillämpligt).
- Start av applikation som flera nedlåsta användare (där tillämpligt).
- Påloggning av en icke-tidigare påloggad användare på systemet
  - a. Start av applikationen av ovanstående användare
- Avinstallation av applikationen i system context.
- Alla applikationer ska testas på Windows 10 oavsett vilket operativs som står i beställningen. Detta är för att slippa testa om alla applikationer senare.
- Detta resultat ska lagras i ePaket)

Paket som driftsätts och därmed tillgängliggörs via eKlient ska ha genomgått UAT där avsedd verksamhet godkänt paketets funktionalitet. Detta resultat ska dokumenteras i ePaket. I de fall paket hämtas ned och genomgår UAT hos annat än den verksamhet som paketerade applikationen, ska även dessa resultat ifrån UAT dokumenteras i ePaket för respektive paket.



## 8. Bilagor

### 8.1 Språkkoder

Exempel på språkkoder enligt ISO 639-1:

SV Svenska

EN Engelska

DE Tyska

FI Finska

DA Danska

FR Franska

ES Spanska

NO Norska

MUL MultiLanguage (eKlientstandard, ingår inte i ISO 639-1)

### 8.2 Medlemskoder

Dessa återfinns numera på eKlient portalen





## 9. Appendix A

### 9.1 Programvara

eKlient rekommenderar att man antingen använder sig av Flexera AdminStudio eller Advanced Installer som huvudverktyg vid MSI-paketering. För enklare editiering av MSI/MST kan även Microsoft Orca och TNE Enterprises InstEdit användas. I de fall leverantör tillhandahåller verktyg för att konfigurera installationsmedia ska dessa verktyg användas.

App-V-paket kan antingen skapas via Flexera AdminStudio med Virtualization Pack, Advanced Installer eller med Microsoft Application Virtualization Sequencer version 5.x. Samma sak gäller för konvertering av MSI till App-V.

MSIX kan skapas exempelvis med hjälp av Flexera AdminStudio, Advanced Installer eller Microsoft MSIX Packaging Tool som finns i Windows Store och är gratis.