

Gör organisationens information agent-ready

En steg-för-steg-instruktion för att tvätta information, bygga en hållbar kontextarkitektur och testa en AI-agent i liten skala

Pilotens grundidé

Välj ett avgränsat användningsområde. Låt AI hjälpa till att inventera, klassificera, rensa och strukturera informationen. Låt verksamheten avgöra vad som är sant, giltigt, normerande och tillåtet. Bygg därefter en agent som får minsta möjliga relevanta kontext och mät både kvalitet och kostnad.

Rekommenderad omfattning

- 4-6 veckor
- Ett tydligt användningsfall
- Ett informationsområde, exempelvis HR, produktdata, kundservice eller avtal
- Cirka 500-2 000 relevanta dokument eller motsvarande datakällor
- 100-200 testfrågor med förväntade svar och källor

Viktigt

Kopiera de dokument och datakällor som ska ingå i piloten till en **säker och tydligt avgränsad arbetsyta**. Arbeta på kopior, inte direkt mot organisationens ordinarie produktionsmiljö eller originaldata.

Syftet är att kunna testa rensning, klassificering, metadata, behörigheter och agentåtkomst utan att riskera att påverka verksamhetskritisk information. Säkerställ också att endast behöriga personer och system får tillgång till materialet, och exkludera känsliga personuppgifter eller annan skyddsvärd information om de inte uttryckligen behövs för piloten.

1. Vad piloten ska bevisa

Målet är inte att "koppla AI till alla våra dokument". Målet är att bevisa att organisationen kan skapa ett begränsat informationsområde där en agent hittar rätt fakta, förstår vad som gäller, respekterar behörigheter och kan arbeta med rimlig token- och API-kostnad.

Framgångskriterium

En bra pilot är inte den som kan svara på flest frågor. Det är den som konsekvent hämtar rätt och aktuell information, kan visa källan, avstår när underlaget är otillräckligt och använder så lite kontext som uppgiften kräver.

Fyra mål för piloten

- Informationskvalitet: identifiera och minska dubletter, överlapp, föråldrade versioner, konflikter och informationsluckor.
- Agent-ready data: göra centrala fakta explicita, märkta, identifierbara och kopplade till tydliga källor och ägare.
- Agentarkitektur: separera kunskap från handling och styra vilken kontext och vilka verktyg agenten får i varje situation.
- Ekonomi: mäta tokenåtgång, retrieval-kostnad, modellkostnad och kostnad per löst uppgift innan lösningen skalas.

Pilotens styrprincip

AI får gärna göra grovjobbet. Verksamheten måste fatta besluten. AI kan hitta konflikter, föreslå metadata och gruppera liknande dokument, men den kan normalt inte veta vilken policy ledningen faktiskt har beslutat om eller vilken källa som ska vara normerande.

Roller och ansvar

Roll	Huvudansvar	Ska besluta om
Pilotägare	Avgränsning, mål och budget	Användningsfall, risknivå, go/no-go
Informationsägare	Kvalitet och aktualitet i innehållet	Vad som är sant, giltigt och normerande
AI/IT-team	Ingestion, sök, metadata, modeller och verktyg	Teknisk arkitektur och mätning
Säkerhet/juridik	Behörighet, sekretess och regelefterlevnad	Vad agenten får se och göra
Slutanvändare	Testar verkliga frågor och arbetsflöden	Om lösningen faktiskt hjälper i arbetet

2. Pilotens arbetsgång

Följ stegen i ordning. Varje steg ska lämna efter sig ett konkret besluts- eller informationsunderlag. Hoppa inte direkt till chatboten. De största kvalitetsvinsterna uppstår ofta före själva modell-anropet.

Steg 1: Välj ett avgränsat användningsfall

Resultat: En definierad uppgift, målgrupp och risknivå.

- Välj ett område där det finns tydliga frågor och relativt tydliga rätta svar, till exempel resepolicy, produktinformation, support eller standardavtal.
- Beskriv exakt vad agenten ska kunna svara på och vilka handlingar den eventuellt ska kunna utföra.
- Definiera vad agenten uttryckligen inte får göra i piloten.

Exempel

"Agenten ska kunna besvara de 100 vanligaste frågorna om tjänsteresor och hänvisa till aktuell policy. Den får inte boka resor, godkänna kostnader eller ge juridisk rådgivning."

Steg 2: Samla och inventera informationsområdet

Resultat: En karta över källor, dokumenttyper, volym, ägare och risker.

Låt AI göra en första inventering av materialet. Den kan klassificera dokument, extrahera metadata och skapa kluster av innehåll som verkar handla om samma sak.

- Samla de relevanta källorna: dokumentbibliotek, intranät, databaser, CRM/ERP, FAQ, API:er och andra verksamhetssystem.
- Låt AI föreslå ämne, dokumenttyp, datum, version, målgrupp, ägare och möjliga frågor som varje källa kan besvara.
- Markera material som saknar ägare, tydlig status eller granskningsdatum.

Steg 3: Tvätta informationen med AI-stöd

Resultat: En åtgärdslista för dubletter, överlapp, versioner, konflikter och känslig information.

- Dubblettrensning: hitta identiska och nästan identiska filer.
- Överlappsrensning: hitta innehåll som återkommer i flera dokument med olika formuleringar.
- Versionsrensning: skilj gällande material från gamla versioner och utkast.
- Konfliktanalys: identifiera olika belopp, datum, ansvar, villkor och regler för samma fråga.
- Aktualitetskontroll: flagga gamla systemnamn, utgångna datum, brutna länkar och andra tecken på inaktualitet.
- Säkerhetskontroll: identifiera personuppgifter, sekretess, API-nycklar och annan information som inte ska ingå i agentens kunskapsbas.

Viktig regel

AI ska normalt inte automatiskt lösa motsägelser genom att "skriva ihop" flera källor. Konflikten ska eskaleras till en informationsägare som avgör vilken uppgift som gäller.

Steg 4: Utse auktoritativa källor och informationsägare

Resultat: En tydlig "source of truth" för varje kritisk uppgift.

- Bestäm var varje kritisk uppgift ska hämtas från. Pris bör exempelvis komma från pris-/ERP-systemet, inte från en gammal presentation.
- Utse en informationsägare för varje viktig källa eller informationsdomän.
- Sätt status: utkast, godkänd, gällande, ersatt, arkiverad.
- Sätt giltighetsperiod och nästa granskningsdatum.

3. Gör informationen agent-ready

När informationen är tvättad ska den struktureras så att agenten inte behöver gissa. Målet är att flytta kritiskt sammanhang från människans huvud till explicita fält, relationer och regler som systemet kan använda.

1. En uppgift, en auktoritativ källa

Bestäm vilken källa som gäller när flera versioner finns. Alternativa källor kan finnas kvar som historik men ska inte konkurrera i normal retrieval.

2. Gör viktiga fakta explicita

Lägg pris, datum, status, villkor, marknad och andra kritiska fakta i strukturerade fält när det är möjligt.

3. Ge data stabila identiteter

Använd stabila ID:n för kunder, produkter, avtal, processer och informationsobjekt så att agenten inte behöver gissa om två namn avser samma sak.

4. Märk upp tid, status och giltighet

Gör det maskinläsbart vad som är aktuellt, godkänt, ersatt eller arkiverat och när informationen gäller.

5. Bygg relationer mellan data

Beskriv hur exempelvis kund, avtal, produkt, prislista, policy och ansvarig funktion hänger ihop.

6. Separera kunskap från handling

Skilj resurser som agenten får läsa från verktyg/API:er som agenten får använda för att genomföra något.

7. Ge agenten minsta relevanta kontext

Filtrera först på domän, behörighet, status, tid och andra metadata. Hämta sedan bara de textstycken och fakta som uppgiften kräver.

8. Bevara källa, behörighet och ansvar

Varje informationsobjekt bör bära med sig ursprung, ägare, åtkomstregler och möjlighet att spåra tillbaka till originalet.

Minsta metadata för pilotens kunskapsobjekt

Fält	Varför agenten behöver det
Objekt-ID	Stabil identitet och spårbarhet
Titel/ämne	Sökning och snabb förståelse
Status	Utkast, godkänd, gällande, arkiverad
Giltig från/till	Tidsfiltrering och historik
Senast verifierad	Aktualitetskontroll

Fält	Varför agenten behöver det
Informationsägare	Ansvar och eskalering
Auktoritativ källa	Prioritering när flera källor finns
Behörighetsklass	Vad agenten får exponera
Relationer	Koppling till kund, avtal, produkt, process etc.
Originalkälla/URL	Källhänvisning och revision

4. Bygg kontextarkitekturen för kvalitet och låg kostnad

Agenten ska inte börja med att läsa hela SharePoint. Den ska börja med den mest precisa och billigaste informationsformen och bara eskalera till mer text och mer avancerad modell när uppgiften kräver det.

Lager	Innehåll	Primär åtkomst	Princip
1. Fakta	Pris, status, saldo, datum, ID, avtalsfält	API/databas	Hämta exakt värde först
2. Kunskap	Policyer, manualer, instruktioner, FAQ	RAG/sökindex	Hämta relevanta kunskapsenheter
3. Situationskontext	Historik, möten, mejl, analyser	Selektiv retrieval	Hämta endast när uppgiften kräver det
4. Handling	Skapa, boka, ändra, skicka, beställa	Verktyg/API/MCP	Separat behörighet och kontroll

Retrieval före resonemang

1. Klassificera frågan: vilket område och vilken uppgift handlar den om?
2. Filtrera på behörighet, status, giltighet, marknad, kund, produkt eller andra relevanta metadata.
3. Hämta strukturerade fakta via API/databas när sådana finns.
4. Hämta därefter ett litet antal relevanta kunskapsenheter via sökning/RAG.
5. Skicka bara den kontext som faktiskt behövs till språkmodellen.
6. Använd en mer avancerad och dyrare modell först när uppgiften kräver djupare resonemang.

Kostnadsprincip

Varje token som inte behövs är både en kostnad och potentiellt brus. Stora kontextfönster gör det möjligt att skicka mycket information, men de gör det inte automatiskt klokt att göra det.

Mät kostnad på uppgiftsnivå

- Inputtokens per fråga och per löst uppgift.
- Outputtokens per fråga.
- Antal retrieval-anrop och verktygsanrop.
- Modellmix: hur stor andel kan hanteras av billig klassificering/extraktion kontra avancerat resonemang?
- Andel återanvändbar/cachningsbar stabil kontext.

- Kostnad per korrekt löst uppgift, inte bara kostnad per API-anrop.

5. Testa innan agenten får förtroende

Bygg testmaterialet samtidigt som ni bygger kunskapsbasen. En agent som inte kan mätas går inte att förvalta.

Steg 5: Skapa ett golden set

Resultat: 100-200 verkliga frågor med förväntade svar, källa och önskat beteende.

- Vanliga frågor med entydigt svar.
- Frågor som kräver två eller flera källor.
- Frågor där gamla och nya versioner finns samtidigt.
- Frågor där källor motsäger varandra.
- Frågor där information saknas och agenten ska avstå.
- Frågor där användaren saknar behörighet.
- Frågor som skulle leda till en handling om agenten hade verktygsåtkomst.

Steg 6: Mät retrieval separat från språkmodellen

Resultat: En tydlig bild av om felet beror på informationen, sökningen eller modellen.

Mätetal	Fråga	Exempel på mål i pilot
Retrieval precision	Hämtades rätt källa?	>90 % på prioriterade frågor
Aktualitet	Prioriterades gällande version?	100 % på kritiska policyfrågor
Groundedness	Stöds svaret av källorna?	>95 %
Avstå korrekt	Avstår agenten när underlag saknas?	>90 %
Behörighet	Blockeras otillåten information?	100 % på definierade testfall
Kostnad	Hur mycket kostar ett korrekt svar?	Baslinje + mål om reducering

Steg 7: Testa agenthandlingar med hårdare krav

Resultat: Bevisa att agenten kan stoppa, fråga och eskalera innan den får skriva eller ändra i verksamhetssystem.

- Kräv tydlig verktygsbehörighet separat från läsbehörighet.
- Testa felaktiga, ofullständiga och manipulerade instruktioner i dokument.
- Inför mänskligt godkännande för ekonomiskt, juridiskt eller verksamhetskritiskt känsliga handlingar.
- Logga vilka källor, instruktioner och verktyg som låg bakom varje kritisk handling.

6. Använd AI som informationsrevisor

När piloten fungerar ska informationen inte lämnas åt sitt öde. AI kan användas löpande för att upptäcka förändringar och avvikelser, men besluten ska fördelas efter risk.

Nivå	AI:s roll	Exempel
Grönt - automatisera	AI kan utföra och logga uppgiften	OCR, formatnormalisering, exakta dubletter, brutna länkar, enkla ämnestaggar
Gult - AI föreslår	Människa granskar eller godkänner	Semantiska dubletter, metadata, aktualitetsflagga, sammanfattning, känslighetsklassning
Rött - verksamheten beslutar	AI får bara presentera underlag	Vilken policy som gäller, juridisk tolkning, radering av original, behörighetsregler, affärskritiska villkor

Steg 8: Sätt en förvaltningsloop

Resultat: En återkommande kontroll av kvalitet, aktualitet, säkerhet och kostnad.

- Flagga dokument som passerat granskningsdatum.
- Upptäck nya konflikter och överlapp efter varje större uppdatering.
- Synkronisera borttagna och ersatta dokument även från sökindex och cache.
- Följ vilka frågor som ofta misslyckas eller kräver onödigt stor kontext.
- Mät kostnad och kvalitet över tid och optimera retrieval, metadata och modellval innan ni ökar modellstorleken.

Förslag: sex veckors pilot

Vecka	Fokus	Leverans
1	Avgränsning och inventering	Use case, källkarta, risknivå, baslinje
2	AI-stödd datatvätt	Dubletter, konflikter, versioner, känslighet
3	Agent-ready struktur	Auktoritativa källor, metadata, ID:n, relationer
4	Kontextarkitektur	Retrieval, filter, API:er, verktyg, behörighet
5	Test och kostnad	Golden set, evals, token- och API-mätning
6	Pilotdrift och beslut	Användartest, avvikelser, go/no-go och skalplan

7. Pilotens leverabler och beslutspunkt

Piloten är klar först när organisationen kan visa både en fungerande agent och en fungerande informationsförvaltning runt agenten.

- [] Avgränsat användningsfall och riskklassning
- [] Inventering över källor och informationsägare

- [] Lista över dubletter, överlapp, gamla versioner och konflikter
- [] Beslut om auktoritativa källor
- [] Metadata- och ID-modell för agent-ready data
- [] Regler för status, giltighet, behörighet och spårbarhet
- [] Kontextarkitektur: fakta, kunskap, situationskontext och handling
- [] Retrieval-strategi med metadatafilter och tydlig källprioritering
- [] Golden set med 100-200 frågor och förväntade svar
- [] Mätetal för kvalitet, säkerhet och kostnad
- [] Förvaltningsrutin för uppdatering, synkronisering och återtest
- [] Go/no-go-beslut för nästa steg

Go/no-go-frågan

Kan vi visa att agenten oftast hittar rätt information, vet vilken version som gäller, respekterar behörigheter, avstår när underlaget är osäkert och gör detta till en kostnad som är rimlig för uppgiften?

Tre slutsatser att bära med till nästa steg

- Datakvalitet är en del av agentens beteende. Dålig information blir inte bra för att modellen är avancerad.
- Agent-ready data minskar både kvalitetsrisk och kostnad eftersom agenten behöver mindre kontext och mindre tolkning.
- Informationsförvaltning blir en kontinuerlig AI-funktion: tvätta, märka, testa, övervaka och förbättra.

Källor och vidare läsning

[Microsoft Azure Architecture Center - RAG: design and evaluation guide](#)

[Microsoft Azure Architecture Center - RAG chunking phase](#)

[Microsoft Azure Architecture Center - RAG chunk enrichment phase](#)

[OWASP Cheat Sheet Series - RAG Security](#)

[Model Context Protocol - 2026-07-28 specification release](#)

Obs: Målnivåerna i pilotens tabeller är exempel och ska kalibreras efter användningsfall, risknivå och verksamhetens krav. De är inte generella branschstandarder.

Anders J Larsson

anders@andersochmia.se

0721-773397