3.4 Tickets vollständig erfassen und kategorisieren

> **Kurz erklärt**
>
> Ein Ticket dokumentiert einen Kontakt, eine Störung, eine Anfrage oder eine andere zu bearbeitende Arbeitseinheit.
>
> Ein gutes Ticket macht nachvollziehbar:
>
> - wer Unterstützung benötigt,
> - welcher Service betroffen ist,
> - was beobachtet wurde,
> - welche Auswirkungen bestehen,
> - seit wann die Situation besteht,
> - was bereits geprüft oder durchgeführt wurde,
> - wer aktuell verantwortlich ist,
> - welcher nächste Schritt notwendig ist,
> - und welches Ergebnis erreicht wurde.
>
> Die Kategorisierung ordnet den Vorgang so ein, dass:
>
> - die richtige Bearbeitung gefunden,
> - geeignetes Wissen angezeigt,
> - ähnliche Vorgänge erkannt,
> - Berichte erstellt,
> - und Verbesserungsmöglichkeiten sichtbar
>
> werden können.
>
> ITIL schreibt kein universelles Ticketformular, kein verpflichtendes Kategoriemodell und keinen für alle Organisationen identischen Ticketstatus vor.
>
> Felder, Kategorien und Arbeitsabläufe müssen zu den Services, Wertströmen, Risiken und Informationsbedürfnissen der jeweiligen Organisation passen.

Warum eine gute Ticketqualität wichtig ist

Ein Ticket ist nicht nur eine technische Notiz.

Es unterstützt unter anderem:

- Kommunikation,
- Priorisierung,
- Diagnose,
- Weiterleitung,
- Eskalation,
- Zusammenarbeit,
- Nachvollziehbarkeit,
- Messung,
- Wissensmanagement,
- Problem Management,
- Lieferantensteuerung,
- und Continual Improvement.

Unvollständige oder unklare Tickets führen häufig zu:

- wiederholten Rückfragen,
- unnötigen Übergaben,
- falscher Priorisierung,
- längeren Wartezeiten,
- doppelter Arbeit,
- ungeeigneten Lösungsversuchen,
- fehlender Benutzerkommunikation,
- und schlechter Auswertbarkeit.

Beispiel für ein unzureichendes Ticket:

> „ Drucker geht nicht.

Daraus ist nicht erkennbar:

- welcher Drucker betroffen ist,
- welcher Benutzer Unterstützung benötigt,
- welche Tätigkeit blockiert ist,
- welche Fehlermeldung erscheint,
- ob weitere Benutzer betroffen sind,
- ob ein alternativer Drucker verfügbar ist,
- und was bereits geprüft wurde.

Eine bessere Erfassung könnte lauten:

> „ Seit etwa 08:20 Uhr können alle vier Arbeitsplätze im Warenausgang keine Versandetiketten über Drucker WD-ET-01 ausgeben. Normale Bürodrucker

> funktionieren. Am Etikettendrucker wird „Druckauftrag nicht verfügbar“ angezeigt. Der Drucker ist eingeschaltet und im Netzwerk erreichbar. Die Druckwarteschlange enthält zwölf offene Aufträge. Der Versandprozess ist derzeit unterbrochen und es besteht kein zugelassener Ersatzdrucker.

> **Merke**
>
> Je besser ein Ticket den tatsächlichen Arbeits- und Servicekontext beschreibt, desto weniger muss die nächste bearbeitende Person erneut erfragen.

Ticket ist ein praktischer Sammelbegriff

Der Begriff **Ticket** wird in vielen ITSM-Werkzeugen als allgemeine Bezeichnung für einen dokumentierten Vorgang verwendet.

Je nach Inhalt kann ein Ticket beispielsweise repräsentieren:

- einen Incident,
- einen Service Request,
- eine Frage,
- eine Beschwerde,
- eine Sicherheitsmeldung,
- eine Aufgabe,
- einen Problem Record,
- einen Change,
- oder einen Lieferantenfall.

Diese Vorgangsarten besitzen unterschiedliche Zwecke.

Vorgangsart	Typischer Zweck
Kontakt	Interaktion mit einem Benutzer oder Stakeholder dokumentieren
Incident	negative Serviceauswirkung behandeln und Service wiederherstellen
Service Request	vorgesehene Benutzeranfrage erfüllen
Problem	Ursache oder mögliche Ursache von Incidents untersuchen
Change	eine Änderung bewerten, autorisieren und kontrolliert steuern
Security Incident	mögliches oder bestätigtes Sicherheitsereignis behandeln

Vorgangsart	Typischer Zweck
Aufgabe	abgegrenzte Tätigkeit innerhalb eines größeren Vorgangs durchführen
Lieferantenfall	externe Unterstützung oder vertragliche Leistung anfordern

> **Wichtig**
>
> Nicht jedes Ticket ist ein Incident.
>
> Nicht jede Benutzerfrage ist ein Service Request.
>
> Die richtige Vorgangsart beeinflusst Bearbeitung, Verantwortung, Messung und Abschlusskriterien.

Kontakt, Ticket und zugrunde liegender Vorgang unterscheiden

Ein Benutzerkontakt und ein Ticket sind nicht immer identisch.

Beispiel:

1. Ein Benutzer meldet telefonisch einen Ausfall.
2. Der Service Desk erstellt ein Incident-Ticket.
3. Drei weitere Benutzer melden dasselbe Problem per Chat.
4. Diese Kontakte werden mit dem bereits bestehenden Incident verknüpft.
5. Ein technisches Team erhält eine zugehörige Diagnoseaufgabe.
6. Ein Provider erhält einen verknüpften Lieferantenfall.
7. Nach der Wiederherstellung wird ein Problem Record für die Ursachenanalyse eröffnet.

Es bestehen damit:

- mehrere Benutzerkontakte,
- ein gemeinsamer Incident,
- technische Teilaufgaben,
- ein Lieferantenfall,
- und möglicherweise ein nachgelagertes Problem.

> **Merke**
>
> Nicht jede neue Meldung benötigt einen vollständig unabhängigen Incident.

> Mehrere Kontakte können dieselbe zugrunde liegende Störung betreffen.

Ziel der Ticket-Erfassung

Die Erfassung soll ausreichende Informationen bereitstellen, damit die Organisation:

- das Anliegen versteht,
- die richtige Vorgangsart bestimmt,
- Auswirkungen und Dringlichkeit bewertet,
- geeignete Fähigkeiten einbindet,
- Fortschritt und Kommunikation nachvollzieht,
- das Ergebnis überprüft,
- und relevante Erkenntnisse weiterverwendet.

Das Ziel ist nicht:

- möglichst viele Felder auszufüllen,
- jeden technischen Wert zu speichern,
- oder Benutzer mit einem umfangreichen Formular zu belasten.

> **Grundsatz**
>
> Es sollen so viele Informationen wie notwendig und so wenige wie möglich erfasst werden.

Ein mögliches Mindestset an Ticketinformationen

Abhängig von Vorgangsart und Organisation können folgende Informationen sinnvoll sein:

Information	Zweck
Vorgangsnummer	eindeutige Referenz
Erstellungszeitpunkt	zeitliche Nachvollziehbarkeit
Meldekanal	Herkunft und Kommunikationskontext
meldende Person	Kontakt und Rückfragen
betroffene Person oder Gruppe	tatsächliche Auswirkung bestimmen
betroffener Service	serviceorientierte Einordnung
kurzer Titel	schnelle verständliche Übersicht
Beschreibung	Situation und Beobachtung dokumentieren

Information	Zweck
Beginn	zeitliche Einordnung der Beeinträchtigung
Auswirkung	Umfang und Geschäftsbezug
Dringlichkeit	zeitliche Bedeutung
Priorität	organisatorisch bestimmte Bearbeitungsreihenfolge
Kategorie	Zuordnung und Auswertung
aktueller Status	Bearbeitungsstand
verantwortliche Rolle oder Gruppe	aktuelles Ownership
bisherige Maßnahmen	doppelte Arbeit vermeiden
Ergebnisse der Maßnahmen	Diagnose nachvollziehen
nächster Schritt	Bearbeitung fortsetzen
Kommunikationszeitpunkt	Benutzerinformation sicherstellen
Lösung oder Erfüllung	Ergebnis dokumentieren
Abschlussbestätigung	tatsächliches Outcome prüfen
verknüpfte Vorgänge	Zusammenhänge sichtbar machen

Nicht jede Vorgangsart benötigt alle Felder.

Ein einfacher Informationskontakt benötigt möglicherweise deutlich weniger Daten als ein kritischer Incident.

Meldende Person und betroffener Benutzer unterscheiden

Die Person, die ein Ticket meldet, ist nicht immer selbst betroffen.

Beispiele:

- Eine Führungskraft meldet eine Störung für ihr Team.
- Die Personalabteilung bestellt einen Arbeitsplatz für einen neuen Mitarbeiter.
- Monitoring meldet einen technischen Zustand ohne menschlichen Anrufer.
- Ein Kollege meldet den Verlust des Geräts einer anderen Person.
- Ein externer Kunde meldet eine Störung für mehrere Benutzer.
- Ein Administrator erfasst einen Incident aufgrund eigener Beobachtung.

Deshalb können getrennte Felder sinnvoll sein für:

- meldende Person,
- betroffene Person,
- betroffene Organisationseinheit,
- betroffenen Standort,

- und bevorzugten Kommunikationskontakt.

> **Typischer Fehler**
>
> Das Ticket wird automatisch der meldenden Person zugeordnet, obwohl eine andere Person oder ein ganzer Bereich betroffen ist.

Der betroffene Service gehört in den Mittelpunkt

Eine serviceorientierte Erfassung fragt nicht nur:

> Welche technische Komponente ist defekt?

Sondern zuerst:

> Welcher Service oder welche Arbeitsfähigkeit ist beeinträchtigt?

Beispiel:

Technische Komponente:

> Datenbankserver DB-02

Betroffener Service:

> Warenwirtschaftssystem

Geschäftliche Auswirkung:

> Lager kann keine Warenausgänge buchen.

Die Komponente kann für Diagnose und technische Zuordnung wichtig sein.

Der Service ist jedoch entscheidend für:

- Auswirkungen,
- Kommunikation,
- Serviceziele,
- Stakeholder,
- und Priorisierung.

Service und Configuration Item unterscheiden

Information	Beispiel
betroffener Service	zentraler Dateiservice
betroffene Anwendung	Dokumentenmanagement
Configuration Item	Fileserver FS-03
Asset	physischer Server mit Inventarnummer
Standort	Berlin-Nord
Benutzergruppe	Buchhaltung

Ein Incident kann mehrere Configuration Items betreffen.

Ein einzelnes Configuration Item kann mehrere Services unterstützen.

Deshalb sollte das Ticket nicht ausschließlich anhand der vermuteten Komponente eingeordnet werden.

Einen guten Tickettitel formulieren

Der Titel sollte die Situation kurz und eindeutig zusammenfassen.

Ungeeignete Titel:

- Problem
- Hilfe
- Geht nicht
- Dringend
- Fehler 4711
- Bitte prüfen
- Benutzerproblem
- Server

Bessere Titel:

- Anmeldung am Zeiterfassungssystem nicht möglich

- Zentrale Dateiablage am Standort Nord nicht erreichbar
- Versandetiketten können im Warenausgang nicht gedruckt werden
- Standardsoftware für neuen Arbeitsplatz anfordern
- Verdächtige Anmeldung für Benutzerkonto erkannt
- Wiederherstellung einer gelöschten Projektdatei prüfen

Ein guter Titel enthält häufig:

- betroffenen Service oder Arbeitsbereich,
- beobachtetes Symptom,
- und gegebenenfalls betroffenen Standort oder Umfang.

> **Praxistipp**
>
> Der Titel sollte auch für eine Person verständlich sein, die das Ticket nicht selbst erfasst hat.

Symptom statt vermuteter Ursache dokumentieren

Zu Beginn eines Incidents ist die tatsächliche Ursache häufig unbekannt.

Ungeeignet:

> Firewall blockiert Anwendung.

wenn dies nur vermutet wird.

Besser:

> Anwendung ist vom Standort Süd nicht über HTTPS erreichbar. Andere Standorte sind nach aktueller Prüfung nicht betroffen. Eine Firewall-Regel ist eine mögliche, aber noch nicht bestätigte Ursache.

Die Dokumentation sollte unterscheiden zwischen:

- Beobachtung,
- Messergebnis,
- Vermutung,
- und bestätigter Ursache.

Art	Beispiel
Beobachtung	Benutzer erhält „Verbindung nicht möglich“
Messergebnis	TCP-Verbindung zu Port 443 schlägt fehl
Vermutung	mögliche Blockierung durch Firewall
bestätigte Ursache	Regel FW-182 wurde beim letzten Change nicht übernommen

> **Merke**
>
> Eine früh eingetragene Vermutung darf nicht unbemerkt zur scheinbar bestätigten Tatsache werden.

Fakten und Bewertungen trennen

Ungeeignet:

> Benutzer hat wieder die falsche Anwendung benutzt.

Besser:

> Der Benutzer öffnete die bisher verwendete Anwendungsversion 4.2. Seit dem Change vom 1. August ist Version 5.0 erforderlich. Die bisherige Verknüpfung auf dem Desktop wurde nicht aktualisiert.

Ungeeignet:

> Netzwerk ist instabil.

Besser:

> Zwischen 09:10 Uhr und 09:25 Uhr gingen 18 von 100 ICMP-Paketen verloren. Zwei VoIP-Gespräche wurden in diesem Zeitraum unterbrochen.

Sachliche Dokumentation verbessert:

- Zusammenarbeit,
- technische Analyse,
- Nachvollziehbarkeit,
- und professionellen Umgang.

Die ursprünglichen Benutzerangaben erhalten

Die Beschreibung des Benutzers kann wichtige Hinweise enthalten.

Beispiel:

> „Beim Speichern erscheint immer die Meldung, dass der Datensatz bereits bearbeitet wird.“

Diese ursprüngliche Beobachtung sollte nicht vollständig durch eine technische Interpretation ersetzt werden.

Eine gute Dokumentation kann enthalten:

- ursprüngliche Benutzerbeschreibung,
- strukturierte Zusammenfassung,
- technische Beobachtungen,
- und durchgeführte Diagnose.

Dadurch bleibt nachvollziehbar:

- was der Benutzer erlebt hat,
- und wie die Organisation die Situation eingeordnet hat.

Zeitangaben vollständig erfassen

Zeitinformationen können für Analyse und Korrelation entscheidend sein.

Hilfreiche Angaben:

- Beginn der Störung,
- Zeitpunkt der Meldung,
- letzter bekannter erfolgreicher Zugriff,
- Zeitpunkt einer Änderung,
- Zeitpunkt einzelner Diagnosemaßnahmen,
- Wiederherstellungszeitpunkt,

- und Abschlusszeitpunkt.

Beispiel:

> Letzter erfolgreicher Zugriff: 08:47 Uhr
> Erste Fehlermeldung: 08:52 Uhr
> Meldung beim Service Desk: 09:03 Uhr
> letzter relevanter Change: 07:30 Uhr

Diese Informationen können mit:

- Monitoring,
- Logs,
- Changes,
- Sicherheitsereignissen,
- und Lieferantenmeldungen

verglichen werden.

> **Typischer Fehler**
>
> Im Ticket steht nur das Erstellungsdatum. Der tatsächliche Beginn der Beeinträchtigung bleibt unbekannt.

Auswirkungen vollständig beschreiben

Die Auswirkungen bestimmen, was der Incident für Benutzer und Organisation bedeutet.

Zu erfassen sind möglicherweise:

- Anzahl betroffener Benutzer,
- betroffene Standorte,
- betroffene Geschäftsprozesse,
- vollständiger oder teilweiser Ausfall,
- verfügbare Zwischenlösung,
- mögliche Sicherheitsauswirkung,
- möglicher Datenverlust,
- Kundenwirkung,
- und zeitkritische Termine.

Unzureichend:

>

> Ein Benutzer betroffen.

Besser:

> Betroffen ist der einzige Arbeitsplatz, über den die Produktionsfreigabe durchgeführt werden kann. Ohne Freigabe steht die Fertigungslinie ab 10:00 Uhr still. Ein Ersatzarbeitsplatz ist nicht verfügbar.

Das Beispiel zeigt:

Die reine Benutzeranzahl beschreibt nicht immer die tatsächliche Auswirkung.

Dringlichkeit nicht nur vom Benutzer übernehmen

Benutzer können ihre gewünschte Dringlichkeit mitteilen.

Die Organisation muss sie jedoch anhand festgelegter Kriterien bewerten.

Ein Benutzer kann beispielsweise „sehr dringend“ auswählen, weil:

- er unter Zeitdruck steht,
- er eine schnelle Bearbeitung erwartet,
- oder die Bedeutung der Auswahl nicht erklärt wurde.

Das Ticket sollte deshalb unterscheiden zwischen:

- vom Benutzer angegebener Dringlichkeit,
- organisatorisch bewerteter Dringlichkeit,
- Auswirkung,
- und daraus bestimmter Priorität.

Die genaue Bewertung wird auf der Seite **3.5 Auswirkungen, Dringlichkeit und Priorität** vertieft.

Die Beschreibung sollte den Normalzustand enthalten

Für die Diagnose ist hilfreich zu wissen:

- Was sollte normalerweise geschehen?
- Was geschieht stattdessen?
- Unter welchen Bedingungen tritt der Fehler auf?

Beispiel:

> Normalerweise wird nach Auswahl von „Auftrag abschließen“ innerhalb weniger Sekunden eine PDF-Rechnung erzeugt. Seit heute 08:30 Uhr bleibt der Ladebildschirm dauerhaft sichtbar. Der Auftrag wird nicht abgeschlossen und es wird keine Fehlermeldung angezeigt.

Diese Beschreibung ist aussagekräftiger als:

> Rechnung funktioniert nicht.

Reproduzierbarkeit dokumentieren

Hilfreiche Angaben:

- tritt der Fehler immer oder nur manchmal auf,
- bei welchen Schritten tritt er auf,
- auf welchen Geräten oder Versionen,
- bei welchen Benutzern,
- mit welchen Daten,
- und unter welchen Bedingungen.

Beispiel:

1. Anwendung öffnen
2. Auftrag 4711 auswählen
3. „PDF exportieren“ anklicken
4. Anwendung schließt sich ohne Fehlermeldung

Zusätzlich:

- tritt bei drei getesteten Aufträgen auf,
- nur in Version 5.2,
- Webversion nicht betroffen.

> **Wichtig**
>
> Reproduktionsschritte dürfen keine vertraulichen Daten oder unsicheren Handlungen unnötig offenlegen.

Bereits durchgeführte Maßnahmen dokumentieren

Das Ticket sollte zeigen:

- welche Maßnahme durchgeführt wurde,
- wer sie durchgeführt hat,
- wann sie durchgeführt wurde,
- warum sie durchgeführt wurde,
- und welches Ergebnis entstand.

Unzureichend:

> Neustart gemacht.

Besser:

> 10:12 Uhr – Anwendungsdienst APP-SVC auf Server APP-02 kontrolliert neu gestartet, nachdem der Prozess nicht mehr auf Health Checks reagierte. Dienst startete erfolgreich. Technischer Health Check ist wieder grün. Benutzeranmeldung schlägt jedoch weiterhin fehl.

Dadurch wird deutlich:

- der Neustart war technisch erfolgreich,
- das ursprüngliche Outcome wurde aber noch nicht erreicht.

Maßnahme und Ergebnis getrennt dokumentieren

Maßnahme	Ergebnis
DNS-Cache geleert	Fehler besteht unverändert
Anwendung neu gestartet	Anmeldung funktioniert einmalig
anderes Benutzerkonto getestet	ebenfalls betroffen
Zugriff über Mobilfunk getestet	funktioniert
Firewall-Regel geprüft	Regel vorhanden und aktiv
Provider kontaktiert	externe Störung bestätigt

Diese Trennung verhindert unklare Notizen wie:

>

> Alles geprüft.

Chronologische Arbeitsnotizen

Eine chronologische Dokumentation unterstützt:

- Schichtübergaben,
- Major-Incident-Koordination,
- Lieferanteneskalation,
- spätere Analyse,
- und Auditierbarkeit.

Beispiel:

> **09:05 Uhr:** Incident durch Service Desk erfasst. Fünf Benutzer betroffen.
> **09:12 Uhr:** Allgemeine Statusseite geprüft, keine bekannte Herstellerstörung.
> **09:18 Uhr:** Anwendung von Standort Nord und Süd getestet. Nur Nord betroffen.
> **09:25 Uhr:** Netzwerkteam eingebunden.
> **09:40 Uhr:** Standorttunnel zeigt Paketverlust. Providerfall P-8821 eröffnet.
> **10:00 Uhr:** Benutzerstatus aktualisiert. Nächstes Update 10:30 Uhr.

Arbeitsnotizen sollten nicht nachträglich so verändert werden, dass wesentliche Entscheidungen oder frühere Zustände nicht mehr nachvollziehbar sind.

Korrekturen sollten transparent erfolgen.

Interne und externe Notizen unterscheiden

Viele Ticketsysteme bieten:

- interne Arbeitsnotizen,
- und für Benutzer sichtbare Kommentare.

Interne Arbeitsnotizen

Können enthalten:

- technische Diagnose,
- Hypothesen,
- interne Koordination,
- sicher freigegebene Protokollauszüge,

- und interne Zuständigkeiten.

Benutzersichtbare Kommentare

Sollten enthalten:

- verständlichen Status,
- Auswirkungen,
- nächste Schritte,
- Handlungshinweise,
- und nächsten Informationszeitpunkt.

> **Sicherheitsrelevant**
>
> Interne Notizen sind nicht automatisch für jede vertrauliche Information geeignet.
>
> Zugriffsrechte, Aufbewahrung und Datenklassifizierung müssen berücksichtigt werden.

Kategorisierung

Kategorisierung ordnet einen Vorgang nach festgelegten Merkmalen ein.

Sie kann unterstützen bei:

- automatischer oder manueller Zuweisung,
- Anzeige passenden Wissens,
- Prioritätsregeln,
- Service-Level-Zuordnung,
- Erkennung ähnlicher Vorgänge,
- Trendanalyse,
- Problem Management,
- und Continual Improvement.

Eine Kategorie sollte einen konkreten Nutzen besitzen.

> **Typischer Fehler**
>
> Kategorien werden nur deshalb angelegt, weil das Werkzeug beliebig viele Ebenen ermöglicht.

Ein einfaches Kategoriemodell

Eine mögliche Struktur lautet:

> Service → Anliegenart → Symptom

Beispiel:

> Dateiablage → Incident → Zugriff nicht möglich

Weitere Beispiele:

- E-Mail → Incident → Nachrichtenversand fehlerhaft
- Arbeitsplatz → Service Request → Standardsoftware anfordern
- Benutzerkonto → Incident → Konto gesperrt
- Netzwerkzugang → Incident → WLAN-Verbindung unterbrochen
- Berechtigung → Service Request → Gruppenmitgliedschaft beantragen
- Informationssicherheit → Sicherheitsmeldung → verdächtige E-Mail

Die konkrete Struktur wird von der Organisation festgelegt.

Serviceorientierte und technische Kategorien

Serviceorientierte Kategorien

Beispiele:

- E-Mail und Kalender
- Dateiablage
- Arbeitsplatz
- Netzwerkzugang
- Warenwirtschaft
- Personalverwaltung
- Zeiterfassung

Vorteile:

- für Benutzer verständlicher,
- besserer Bezug zu Serviceauswirkungen,
- Unterstützung von Serviceberichten.

Technische Kategorien

Beispiele:

- DNS
- DHCP
- Firewall
- Datenbank
- Betriebssystem
- Speicher
- Zertifikat

Vorteile:

- hilfreich für technische Analyse,
- Unterstützung spezialisierter Teams,
- technische Trendanalyse.

Beide Perspektiven können kombiniert werden.

> **Praxistipp**
>
> Der Benutzer wählt möglichst den betroffenen Service oder sein Anliegen.
>
> Die technische Einordnung kann während der Bearbeitung ergänzt werden.

Benutzerkategorie und interne Kategorie unterscheiden

Ein Portal sollte nicht zwingend dieselben Kategorien anzeigen wie das interne Ticketsystem.

Benutzerfreundliche Auswahl:

- Ich kann mich nicht anmelden.
- Ich benötige Zugriff.
- Eine Anwendung funktioniert nicht.
- Ich habe eine verdächtige E-Mail erhalten.
- Ich möchte Software bestellen.

Interne Zuordnung:

- Service: Identity and Access
- Vorgangsart: Incident
- Symptom: MFA fehlgeschlagen
- technische Domäne: Entra ID
- Supportgruppe: Identity Operations

Dadurch muss der Benutzer keine interne IT-Struktur kennen.

Mehrere Kategorisierungsdimensionen verwenden

Eine einzige hierarchische Kategorie kann schnell zu groß und unübersichtlich werden.

Alternativ können mehrere getrennte Merkmale verwendet werden.

Dimension	Beispiel
Service	E-Mail
Vorgangsart	Incident
Symptom	Versand nicht möglich
technische Domäne	Mail Gateway
Standort	Berlin
Benutzergruppe	Vertrieb
Sicherheitsbezug	nein
Ursachencode	Zertifikat abgelaufen
Lösungscode	Zertifikat erneuert

Vorteile:

- bessere Auswertung,
- weniger tiefe Kategorien,
- und flexiblere Zuordnung.

Nachteile:

- mehr Felder,
- höherer Pflegeaufwand,
- und mögliche uneinheitliche Eingaben.

Deshalb sollten nur tatsächlich benötigte Dimensionen verwendet werden.

Symptom-, Ursachen- und Lösungskategorien unterscheiden

Diese Kategorien entstehen zu unterschiedlichen Zeitpunkten.

Kategorie	Zeitpunkt	Beispiel
Symptomkategorie	bei Erfassung	Anmeldung nicht möglich
technische Kategorie	während Diagnose	Identitätsdienst

Kategorie	Zeitpunkt	Beispiel
Ursachenkategorie	nach Bestätigung	Zertifikat abgelaufen
Lösungskategorie	nach Wiederherstellung	Zertifikat erneuert

Zu Beginn darf nicht so kategorisiert werden, als wäre die Ursache bereits bekannt.

> **Typischer Fehler**
>
> Das Ticket wird bei der ersten Meldung als „Netzwerkfehler“ kategorisiert und bleibt dort, obwohl später ein abgelaufenes Anwendungstoken als Ursache bestätigt wird.

Kategorien dürfen geändert werden

Eine erste Kategorisierung ist häufig vorläufig.

Während der Bearbeitung können neue Erkenntnisse entstehen.

Eine Kategorie sollte geändert werden, wenn:

- der betroffene Service falsch erkannt wurde,
- die Vorgangsart falsch war,
- eine technische Ursache bestätigt wurde,
- oder eine genauere Einordnung möglich ist.

Die Änderung sollte:

- nachvollziehbar,
- nach festgelegten Regeln,
- und ohne Verlust der ursprünglichen Informationen

erfolgen.

Kategorie „Sonstiges“ kontrolliert verwenden

Eine Kategorie wie:

- Sonstiges,
- Allgemein,
- Unbekannt,
- oder Andere

kann notwendig sein.

Sie darf jedoch nicht zur dauerhaften Standardkategorie werden.

Eine hohe Nutzung kann anzeigen:

- Kategorien sind unverständlich,
- wichtige Services fehlen,
- Mitarbeiterschulung ist unzureichend,
- das Portal ist schlecht gestaltet,
- oder neue Bedarfsmuster sind entstanden.

> **Praxistipp**
>
> Vorgänge in „Sonstiges“ sollten regelmäßig ausgewertet und bei Bedarf neu kategorisiert werden.

Nicht zu viele Kategorieebenen verwenden

Ungeeignetes Beispiel:

> IT → Infrastruktur → Netzwerk → Standortnetz → LAN → Switch → Cisco → Modell → Port → Fehler

Eine solche Struktur kann:

- langsam,
- fehleranfällig,
- schwer wartbar,
- und für Benutzer ungeeignet

sein.

Eine einfachere Kombination kann ausreichen:

- Service: Standortnetzwerk
- Symptom: Verbindung unterbrochen
- betroffenes CI: Switch SW-N-03
- technische Gruppe: Network Operations

> **Grundsatz**
>
> Die Kategorie sollte die Arbeit unterstützen und nicht selbst zu einer zusätzlichen komplizierten Arbeit werden.

Kategorien nicht nach Teams aufbauen

Eine Kategorie wie:

- First Level
- Netzwerkteam
- Serverteam
- Anwendungsteam

beschreibt organisatorische Zuständigkeit, aber nicht unbedingt den Service oder das Anliegen.

Teams können sich verändern.

Services und Benutzerbedürfnisse bleiben häufig stabiler.

Besser ist:

- Service und Anliegen kategorisieren,
- zuständige Gruppe daraus ableiten,
- und Zuweisungsregeln unabhängig pflegen.

Vorgangsart richtig bestimmen

Eine wichtige erste Einordnung ist die Unterscheidung zwischen verschiedenen Vorgangsarten.

Incident

Ein Incident liegt vor, wenn eine ungeplante negative Auswirkung auf einen Service behandelt werden muss.

Beispiele:

- Anwendung ist nicht erreichbar.
- vorhandener Zugriff funktioniert nicht.
- Druckservice liefert keine Aufträge.
- Leistung ist erheblich vermindert.
- vorhandene Funktion erzeugt Fehler.

Service Request

Ein Service Request ist eine vorgesehene Benutzeranfrage.

Beispiele:

- Standardsoftware anfordern
- neuen Zugriff beantragen
- Information zu einem Service erhalten
- Arbeitsplatzgerät bestellen
- Verteilerliste ändern lassen

Frage oder Informationskontakt

Beispiele:

- Wann findet die Wartung statt?
- Welche Software ist freigegeben?
- Wie erreiche ich den Service außerhalb der Geschäftszeit?

Eine Frage kann möglicherweise direkt beantwortet werden, ohne einen umfangreichen Request-Wertstrom zu starten.

Sicherheitsmeldung

Beispiele:

- verdächtige E-Mail
- verlorenes Gerät
- möglicher unberechtigter Zugriff
- versehentliche Datenoffenlegung
- ungewöhnliches Anmeldeverhalten

Eine Sicherheitsmeldung benötigt möglicherweise:

- besondere Zugriffsrechte,
- vertrauliche Bearbeitung,
- und schnelle Weiterleitung an das Sicherheitsteam.

Beschwerde oder Feedback

Beispiele:

- zugesagte Rückmeldung wurde nicht eingehalten
- Portal ist nicht barrierefrei nutzbar

- wiederkehrende Störung beeinträchtigt die Arbeit
- Servicebeschreibung ist unverständlich

Feedback kann mit einem Incident oder Request verbunden sein, besitzt aber möglicherweise einen eigenen Bearbeitungs- und Verbesserungsweg.

Fehleinordnungen korrigieren

Beispiel:

Ein Benutzer wählt:

> Service Request – Kennwort ändern

Während der Erfassung wird erkannt:

- Das Konto ist aufgrund eines Fehlers im Identitätsdienst gesperrt.
- Mehrere Benutzer sind betroffen.
- Die normale Selbstverwaltung funktioniert nicht.

Der Vorgang sollte dann möglicherweise als Incident behandelt werden.

Ein anderes Beispiel:

Der Benutzer meldet:

> VPN funktioniert nicht.

Die Analyse zeigt:

- VPN ist grundsätzlich funktionsfähig.
- Der Benutzer besitzt noch keine genehmigte VPN-Berechtigung.

Dies kann ein Service Request statt eines Incidents sein.

> **Merke**
>
> Die Formulierung oder Portalwahl des Benutzers bestimmt nicht allein die endgültige Vorgangsart.

Duplikate erkennen

Bei größeren Störungen entstehen häufig mehrere Tickets zum selben Incident.

Duplikaterkennung verhindert:

- parallele Diagnose,
- widersprüchliche Maßnahmen,
- mehrfache Lieferantenmeldungen,
- und unklare Statuskommunikation.

Vor dem Erstellen eines neuen Incidents kann geprüft werden:

- Ist bereits eine bekannte Störung veröffentlicht?
- Existiert ein Incident für denselben Service?
- Sind Zeitraum, Standort und Symptom identisch?
- Besteht ein gemeinsamer technischer Auslöser?
- Handelt es sich tatsächlich um denselben oder nur um einen ähnlichen Fehler?

Duplikate nicht einfach löschen

Ein Duplikat kann wichtige Informationen enthalten:

- zusätzliche betroffene Benutzer,
- weiteren Standort,
- abweichendes Symptom,
- neue Zeitangabe,
- oder besondere Auswirkung.

Eine sinnvolle Behandlung kann sein:

1. Kontaktinformationen erhalten
2. Ticket mit dem Haupt-Incident verknüpfen
3. betroffene Person oder Gruppe ergänzen
4. Benutzerkommunikation mit dem Haupt-Incident abstimmen
5. Duplikat nachvollziehbar schließen oder unterordnen

Parent-, Child- und Sammel-Incidents

Bei einem größeren Incident kann ein Hauptvorgang verwendet werden.

Haupt-Incident

Enthält:

- technische Gesamtkoordination,

- zentrale Auswirkung,
- gemeinsame Statusinformationen,
- und Wiederherstellungsstand.

Verknüpfte Meldungen

Enthalten:

- einzelne betroffene Benutzer,
- lokale Besonderheiten,
- individuelle Rückfragen,
- und Bestätigung der Wiederherstellung.

Das konkrete Modell hängt vom Ticketsystem und der Organisation ab.

Verknüpfungen zwischen Vorgängen

Tickets können miteinander verbunden werden.

Beispiele:

- Incident wurde durch Change verursacht.
- mehrere Incidents gehören zu einem Problem.
- Service Request benötigt einen Change.
- Incident besitzt einen Lieferantenfall.
- Major Incident besitzt mehrere technische Aufgaben.
- Sicherheitsmeldung führt zu einem Security Incident.
- Knowledge-Artikel dokumentiert eine bekannte Lösung.
- Verbesserungsidee entsteht aus einer Incident-Auswertung.

Mögliche Beziehungstypen:

- verursacht durch
- betrifft
- Duplikat von
- gelöst durch
- gehört zu
- Folge von
- blockiert durch
- bekanntes Problem
- zugehöriger Change
- zugehöriger Knowledge-Artikel

> **Wichtig**

> Verknüpfungen sollten eine erkennbare Bedeutung besitzen.
>
> Eine große Menge unklarer Beziehungen verbessert die Nachvollziehbarkeit nicht.

Status eines Tickets

Ein Status zeigt den aktuellen Bearbeitungszustand.

Mögliche Status sind beispielsweise:

- Neu
- Erfasst
- In Prüfung
- In Bearbeitung
- Warten auf Benutzer
- Warten auf Lieferant
- Warten auf Change
- Gelöst
- Abgeschlossen
- Abgebrochen

ITIL schreibt kein universelles Statusmodell vor.

Die Organisation muss festlegen:

- was jeder Status bedeutet,
- wer ihn setzen darf,
- welche Verantwortung weiterhin besteht,
- ob Zielzeiten weiterlaufen,
- und welche Kommunikation ausgelöst wird.

„Warten“ benötigt einen Grund und nächsten Schritt

Unzureichend:

> “ Status: Warten

Besser:

>

> Warten auf Benutzer – benötigt wird ein Screenshot der genauen Fehlermeldung. Rückmeldung bis 5. August erbeten. Falls keine Antwort erfolgt, wird am 6. August ein weiterer Kontaktversuch durchgeführt.

Oder:

> Warten auf Lieferant – Providerfall P-8821 wurde um 10:40 Uhr eröffnet. Nächste vertragliche Rückmeldung wird bis 12:40 Uhr erwartet. Interner Ansprechpartner bleibt Network Operations.

> **Typischer Fehler**
>
> Ein Ticket wird auf „Warten“ gesetzt, ohne dass erkennbar ist, worauf gewartet wird und wer den Vorgang wieder aufnimmt.

Gelöst und abgeschlossen unterscheiden

Eine Organisation kann zwischen:

- technisch oder fachlich gelöst,
- und formal abgeschlossen

unterscheiden.

Gelöst

- Lösung oder Erfüllung wurde durchgeführt.
- Ergebnis wird möglicherweise noch bestätigt oder beobachtet.

Abgeschlossen

- Abschlusskriterien wurden erfüllt.
- Dokumentation ist vollständig.
- offene Folgeaktivitäten sind zugeordnet.
- Benutzer wurde informiert.
- Vorgang benötigt keine weitere aktive Bearbeitung.

Dieses Modell ist möglich, aber nicht verpflichtend.

Automatischer Abschluss

Ein Ticket kann nach einer festgelegten Frist automatisch geschlossen werden.

Dabei sollte berücksichtigt werden:

- Wurde der Benutzer verständlich informiert?
- War die Frist angemessen?
- Besteht ein einfacher Weg zur Wiederaufnahme?
- Ist der Service technisch überprüft?
- Handelt es sich um einen kritischen oder sensiblen Vorgang?
- Bestehen offene Folgeaktivitäten?

Automatischer Abschluss darf nicht nur dazu dienen, Kennzahlen zu verbessern.

Ownership und Zuweisung

Ein Ticket sollte jederzeit eine erkennbare Verantwortung besitzen.

Mögliche Felder:

- verantwortliche Gruppe,
- verantwortliche Person,
- Service Owner,
- Incident-Koordinator,
- und Kommunikationsverantwortlicher.

Eine Gruppe allein reicht möglicherweise nicht aus, wenn:

- niemand den Vorgang aktiv übernimmt,
- keine nächste Aktion geplant ist,
- oder mehrere Teams beteiligt sind.

> **Merke**
>
> „Zugewiesen“ bedeutet nicht automatisch „wird bearbeitet“.

Weiterleitung benötigt vollständige Informationen

Vor einer Weiterleitung sollte geprüft werden:

- Ist der betroffene Service bekannt?
- Ist die Auswirkung beschrieben?
- Ist die Priorität nachvollziehbar?
- Wurden geeignete erste Prüfungen durchgeführt?

- Sind Ergebnisse dokumentiert?
- Ist die Zielgruppe richtig?
- Ist klar, welche Unterstützung benötigt wird?
- Bleibt die Benutzerkommunikation geregelt?

Ungeeignete Weiterleitung:

> Bitte prüfen.

Besser:

> Alle Benutzer am Standort Nord können seit 09:15 Uhr den zentralen Dateiservice nicht erreichen. DNS-Auflösung und lokale Netzwerkverbindung funktionieren. Der Zugriff auf TCP-Port 445 schlägt fehl. Standort Süd ist nicht betroffen. Benötigt wird die Prüfung des Standorttunnels und der Firewall. Nächste Benutzerstatusmeldung wurde für 10:30 Uhr angekündigt.

Ticket-Pingpong erkennen

Ticket-Pingpong entsteht, wenn ein Vorgang mehrfach zwischen Gruppen weitergeleitet wird.

Mögliche Ursachen:

- Kategorien sind unklar,
- Verantwortung ist nicht definiert,
- Informationen fehlen,
- Teams lehnen Vorgänge ohne Abstimmung ab,
- Serviceabhängigkeiten sind unbekannt,
- oder die Organisation arbeitet zu stark in Silos.

Geeignete Maßnahmen:

- warme Übergabe,
- gemeinsames Swarming,
- klare Zuweisungsregeln,
- bessere Serviceinformationen,
- und Auswertung häufig weitergeleiteter Tickets.

Anhänge, Screenshots und Protokolle

Anhänge können die Diagnose unterstützen.

Beispiele:

- Screenshot einer Fehlermeldung
- gekürzter Logauszug
- Export technischer Diagnosedaten
- Foto eines Hardwarefehlers
- Konfigurationsauszug
- Testprotokoll

Vor der Speicherung sollte geprüft werden:

- Wird der Anhang tatsächlich benötigt?
- Enthält er personenbezogene Daten?
- Sind Kennwörter, Tokens oder Schlüssel sichtbar?
- Enthält er vertrauliche Geschäftsangaben?
- Besitzen alle Ticketleser die notwendige Berechtigung?
- Wie lange wird der Anhang gespeichert?
- Kann eine kleinere oder anonymisierte Auswahl ausreichen?

Screenshots richtig verwenden

Ein Screenshot sollte möglichst zeigen:

- relevante Fehlermeldung,
- Anwendung oder Kontext,
- Zeitpunkt, falls wichtig,
- und notwendige technische Details.

Er sollte möglichst nicht zeigen:

- private Nachrichten,
- vollständige Kundenlisten,
- Kennwörter,
- Zugriffstokens,
- nicht benötigte personenbezogene Daten,
- oder andere vertrauliche Fenster.

> **Praxistipp**
>
> Fordere möglichst nur den benötigten Bildausschnitt an und erkläre, welche sensiblen Inhalte vorher geschlossen oder entfernt werden sollen.

Protokolldaten sicher behandeln

Logs können enthalten:

- Benutzernamen,
- IP-Adressen,
- Dateipfade,
- Session-IDs,
- Tokens,
- personenbezogene Informationen,
- und interne Architekturdetails.

Sie sollten:

- nur zweckgebunden verwendet,
- angemessen geschützt,
- nicht unkontrolliert in offene Kommentare kopiert,
- und nach den Vorgaben der Organisation aufbewahrt

werden.

Kennwörter und Zugangsdaten gehören nicht ins Ticket

Nicht gespeichert werden sollten insbesondere:

- Kennwörter,
- private Schlüssel,
- API-Secrets,
- vollständige Zugriffstokens,
- Wiederherstellungscodes,
- und ungeschützte Zugangsdaten.

Erhält der Service Desk solche Informationen, müssen die Sicherheitsverfahren der Organisation angewendet werden.

Mögliche Maßnahmen:

- Information aus dem sichtbaren Vorgang entfernen oder schützen,
- betroffene Zugangsdaten widerrufen,
- Benutzer informieren,
- und Sicherheitsrolle einbinden.

Datenschutz und Datenminimierung

Ein Ticket darf nur Informationen enthalten, die für einen legitimen Zweck benötigt werden.

Zu prüfen sind:

- Welche personenbezogenen Daten sind notwendig?
- Wer darf sie sehen?
- Wie lange werden sie gespeichert?
- Müssen Daten anonymisiert werden?
- Sind besondere Kategorien personenbezogener Daten betroffen?
- Ist das Ticket für den gewählten Empfängerkreis geeignet?

Ungeeignet:

> Vollständige Krankheitsdiagnose eines Mitarbeiters im allgemeinen IT-Ticket.

Ausreichend kann je nach Situation sein:

> Benutzer benötigt aufgrund einer genehmigten Arbeitsplatzanpassung eine alternative Eingabemöglichkeit.

Die verbindlichen Regeln ergeben sich aus den Datenschutz- und Sicherheitsvorgaben der Organisation.

Besondere Behandlung von Sicherheitsmeldungen

Sicherheitsbezogene Tickets können besondere Anforderungen besitzen.

Beispiele:

- eingeschränkter Leserkreis,
- separate Sicherheitswarteschlange,
- unveränderte Beweissicherung,
- besondere Aufbewahrung,
- sofortige Eskalation,
- und kontrollierte Kommunikation.

Ein allgemeiner Service-Desk-Mitarbeiter sollte wissen:

- welche Hinweise sicherheitsrelevant sind,
- welche Informationen er erfassen darf,
- was nicht verändert werden darf,
- und an wen der Vorgang weitergegeben wird.

Ticket-Erfassung bei einem möglichen Major Incident

Bei einem möglichen Major Incident sind besonders wichtig:

- genauer Beginn,
- betroffene Services,
- betroffene Standorte und Benutzergruppen,
- Geschäfts- oder Kundenwirkung,
- verfügbare Zwischenlösungen,
- letzte Changes,
- bisherige Maßnahmen,
- beteiligte Teams,
- Entscheidungs- und Kommunikationsverantwortung,
- und nächster Statuszeitpunkt.

Ein kurzer Erstbericht darf zunächst unvollständig sein, wenn schnelles Handeln notwendig ist.

Die Informationen müssen anschließend kontrolliert ergänzt werden.

> **Grundsatz**
>
> Schnelle Erfassung und ausreichende Qualität müssen ausbalanciert werden.
>
> Ein kritischer Incident darf nicht wegen eines umfangreichen Pflichtformulars unnötig verzögert werden.

Pflichtfelder

Pflichtfelder können Informationsqualität verbessern.

Sie können jedoch auch:

- Erfassung verzögern,
- falsche Platzhalter erzeugen,
- Benutzer überfordern,
- oder Mitarbeiter zum Umgehen des Systems verleiten.

Ein Feld sollte verpflichtend sein, wenn die Information:

- regelmäßig benötigt wird,
- zum Erfassungszeitpunkt bekannt sein kann,
- und einen erkennbaren Nutzen besitzt.

Ungeeignetes Pflichtfeld bei der Ersterfassung:

>

> bestätigte technische Ursache

Die Ursache ist zu diesem Zeitpunkt häufig noch unbekannt.

Besser:

> beobachtetes Symptom

Dynamische Formulare

Dynamische Formulare zeigen nur Felder an, die zur gewählten Situation passen.

Beispiel:

Auswahl:

> Verdächtige E-Mail

Zusätzliche Felder:

- Zeitpunkt des Empfangs
- wurde Link geöffnet?
- wurde Anhang geöffnet?
- betroffener Benutzer
- sichere Uploadmöglichkeit

Auswahl:

> Software anfordern

Zusätzliche Felder:

- benötigte Anwendung
- Gerät
- geschäftlicher Zweck
- benötigter Zeitpunkt
- Genehmigung

Dadurch können Formulare kürzer und verständlicher werden.

Vorlagen und Ticketmodelle

Für häufige Situationen können Vorlagen verwendet werden.

Beispiele:

- Benutzerkonto gesperrt
- Softwareanforderung
- Standortausfall
- verlorenes Endgerät
- verdächtige E-Mail
- Druckerstörung
- Datenwiederherstellung
- Lieferanteneskalation

Eine Vorlage kann enthalten:

- notwendige Fragen,
- Standardaufgaben,
- Zuweisungsregeln,
- Kommunikationsvorlagen,
- und Abschlusskriterien.

Vorlagen müssen regelmäßig geprüft werden.

Eine veraltete Vorlage kann falsche Diagnose oder unnötige Schritte fördern.

Automatisierte Erfassung

Tickets können automatisch entstehen durch:

- Monitoring,
- E-Mail,
- Portal,
- Chatbot,
- Endpoint Management,
- Sicherheitsplattform,
- Cloud-Provider,
- oder Schnittstellen.

Bei automatischer Erfassung sollte geklärt sein:

- Ist die Meldung handlungsrelevant?

- Existiert bereits ein Vorgang?
- Welcher Service ist betroffen?
- Welche Kategorie ist zuverlässig bestimmbar?
- Welche Priorität darf automatisch gesetzt werden?
- Wer überprüft die Information?
- Wie werden Fehlalarme behandelt?
- Wann werden Benutzer informiert?

> **Merke**
>
> Automatisch erzeugt bedeutet nicht automatisch richtig kategorisiert oder priorisiert.

KI-gestützte Kategorisierung

KI kann Vorschläge erzeugen für:

- Vorgangsart,
- Kategorie,
- betroffenen Service,
- ähnliche Tickets,
- zuständige Gruppe,
- Priorität,
- und passende Wissensartikel.

Mögliche Vorteile:

- schnellere Einordnung,
- einheitlichere Kategorien,
- weniger manuelle Arbeit,
- und Erkennung ähnlicher Formulierungen.

Mögliche Risiken:

- falsche Zuordnung,
- unerkannte Sicherheitsmeldungen,
- Verzerrungen durch schlechte Trainingsdaten,
- Übernahme veralteter Kategorien,
- und fehlende Nachvollziehbarkeit.

KI-Ausgaben müssen überprüfbar bleiben

Zu klären ist:

- Welche Daten verwendet die KI?
- Welche Kategorien darf sie automatisch setzen?
- Wann ist menschliche Bestätigung erforderlich?
- Wie werden Fehlzuordnungen korrigiert?
- Wie wird aus Korrekturen gelernt?
- Welche sensiblen Daten werden verarbeitet?
- Wer trägt Verantwortung?
- Wie wird die Qualität gemessen?

Bei kritischen oder sicherheitsrelevanten Vorgängen sollte eine ungeprüfte vollautomatische Einordnung vermieden werden.

Kategorien regelmäßig pflegen

Ein Kategoriemodell ist nicht dauerhaft unverändert geeignet.

Änderungen entstehen durch:

- neue Services,
- neue Produkte,
- organisatorische Veränderungen,
- neue Technologien,
- neue Lieferanten,
- und Erfahrungen aus der Bearbeitung.

Regelmäßig zu prüfen sind:

- Welche Kategorien werden kaum verwendet?
- Welche Kategorien werden zu häufig verwendet?
- Wo entstehen viele Fehlzuordnungen?
- Welche Kategorien führen regelmäßig zur falschen Gruppe?
- Wo wird häufig „Sonstiges“ gewählt?
- Welche neuen Muster sind entstanden?
- Sind Begriffe für Benutzer verständlich?
- Sind Berichte weiterhin sinnvoll?

Kategorisierung und Reporting

Kategorien ermöglichen Auswertungen wie:

- häufig betroffene Services,
- wiederkehrende Symptome,
- hohe Incident-Mengen,
- Service-Request-Nachfrage,
- häufige Weiterleitungen,

- Lieferantenabhängigkeiten,
- und Verbesserungspotenzial.

Die Aussagekraft hängt von der Datenqualität ab.

Unzuverlässige Kategorien erzeugen unzuverlässige Berichte.

> **Grundsatz**
>
> Schlechte Eingangsdaten werden durch ein modernes Dashboard nicht automatisch zu guten Entscheidungsinformationen.

Keine falschen Anreize durch Kategorien

Mitarbeiter können Kategorien so wählen, dass:

- Zielzeiten günstiger erscheinen,
- der Vorgang nicht zur eigenen Gruppe zählt,
- oder Kennzahlen besser aussehen.

Beispiele:

- Incident wird als Service Request erfasst.
- hoher Prioritätsfall wird heruntergestuft.
- wiederkehrender Incident wird unter wechselnden Kategorien gespeichert.
- Ticket wird als „Benutzerfehler“ abgeschlossen, ohne die Ursache zu untersuchen.

Governance und Qualitätskontrollen müssen solche Fehlanreize berücksichtigen.

Ticketqualität messen

Mögliche Qualitätsmerkmale:

- betroffener Service eindeutig,
- Titel verständlich,
- Symptom vollständig,
- Auswirkung nachvollziehbar,
- Zeitangaben vorhanden,
- Kategorie geeignet,
- Vorgangsart korrekt,
- Maßnahmen und Ergebnisse dokumentiert,
- Ownership sichtbar,
- nächster Schritt bekannt,

- Benutzerkommunikation nachvollziehbar,
- Lösung vollständig,
- Abschlusskriterien erfüllt,
- und Datenschutz eingehalten.

Eine reine Vollständigkeitsquote reicht nicht aus.

Ein ausgefülltes Feld kann trotzdem:

- falsch,
- unverständlich,
- veraltet,
- oder nutzlos

sein.

Qualitätsprüfung ohne unnötige Bürokratie

Mögliche Verfahren:

- Stichproben,
- Peer Review,
- Coaching,
- automatische Plausibilitätsprüfung,
- Auswertung wiedereröffneter Tickets,
- Analyse häufiger Rückfragen,
- und Prüfung häufig weitergeleiteter Vorgänge.

Die Qualitätsprüfung sollte vor allem:

- Lernen,
- einheitliche Arbeitsweise,
- und Verbesserung

unterstützen.

Sie sollte nicht ausschließlich zur persönlichen Fehlerbewertung verwendet werden.

Praxisbeispiel: unvollständiges Incident-Ticket

Ursprüngliches Ticket

> Titel: Outlook geht nicht
> Kategorie: Sonstiges
> Priorität: Hoch
> Beschreibung: Bitte schnell lösen.

Fehlende Informationen

- betroffener Service
- tatsächliches Symptom
- betroffener Benutzer
- Zeitpunkt
- weitere Betroffene
- Auswirkung
- Fehlermeldung
- bereits durchgeführte Maßnahmen
- Begründung der Priorität

Verbesserte Erfassung

> **Titel:** E-Mail-Versand über Outlook am Standort Süd nicht möglich
> **Service:** E-Mail und Kalender
> **Vorgangsart:** Incident
> **Symptom:** Nachrichten bleiben im Postausgang
> **Beginn:** etwa 09:10 Uhr
> **Betroffene:** aktuell sieben Benutzer am Standort Süd
> **Auswirkung:** externe Nachrichten können nicht versendet werden; Empfang funktioniert
> **Fehlermeldung:** „Verbindung zum Ausgangsserver nicht möglich“
> **Prüfung:** Webmail-Versand funktioniert; Outlook und Geräte wurden nicht verändert
> **Zwischenlösung:** Versand über Webmail möglich
> **Nächster Schritt:** lokale Outlook- und Netzwerkverbindung prüfen

---

Praxisbeispiel: Incident oder Service Request

Benutzermeldung:

> Ich komme nicht auf das Projektlaufwerk.

Möglicher Incident

- Zugriff bestand zuvor.
- Berechtigung ist weiterhin zugewiesen.
- Laufwerk ist plötzlich nicht erreichbar.
- weitere Benutzer sind betroffen.

Möglicher Service Request

- Zugriff bestand noch nie.
- Benutzer benötigt eine neue Projektberechtigung.
- Genehmigung durch Projektverantwortlichen ist erforderlich.

Mögliche Sicherheitssituation

- Benutzer sieht plötzlich Daten eines fremden Projekts.
- unberechtigte Berechtigung muss sofort geprüft werden.

Dieselbe Formulierung kann zu unterschiedlichen Vorgangsarten führen.

Praxisbeispiel: Duplikate bei einer Störung

Innerhalb von zehn Minuten melden 35 Benutzer:

> „ Dateiablage nicht erreichbar.

Geeignete Behandlung:

1. gemeinsamen Incident erkennen
2. betroffene Benutzer und Standorte ergänzen
3. Duplikate mit Haupt-Incident verknüpfen
4. technische Bearbeitung zentral koordinieren
5. allgemeine Statusmeldung veröffentlichen
6. individuelle Sonderauswirkungen gesondert dokumentieren
7. Wiederherstellung für betroffene Gruppen prüfen

Ungeeignet:

- 35 Teams bearbeiten 35 unabhängige Incidents,
- oder alle Meldungen werden gelöscht und wichtige Auswirkungen gehen verloren.

Praxisbeispiel: falsche Kategorisierung

Ticket wurde erfasst als:

> Netzwerk → WLAN → Verbindungsfehler

Die Analyse zeigt:

- Netzwerkverbindung funktioniert.
- Nur eine Anwendung ist betroffen.
- Ursache ist eine abgelaufene Anwendungslizenz.

Korrektur:

- Service: Fachanwendung
- Vorgangsart: Incident
- Symptom: Anmeldung nicht möglich
- technische Ursache: Lizenz abgelaufen
- Lösung: Lizenz erneuert

Die ursprüngliche Benutzerbeschreibung und bisherige Diagnose bleiben erhalten.

Praxisbeispiel: sicherheitskritischer Anhang

Ein Benutzer hängt einen Screenshot an.

Darauf sichtbar:

- Fehlermeldung,
- Benutzername,
- vollständiger Zugriffstoken,
- interne Serveradresse,
- und Kundendaten.

Geeignete Reaktion:

- Zugriff auf den Anhang einschränken,
- Sicherheitsverfahren anwenden,
- Token widerrufen,
- unnötige Daten entfernen oder geschützt behandeln,
- Benutzer über sichere Übermittlung informieren,
- und prüfen, ob eine Datenoffenlegung vorliegt.

Das Ticket darf nicht unverändert an große Verteiler oder externe Lieferanten weitergeleitet werden.

Typische Fehler bei Ticket-Erfassung und Kategorisierung

Fehler 1: Titel ohne Aussage

„Problem“, „Hilfe“ oder „Dringend“ ermöglichen keine schnelle Einordnung.

Fehler 2: vermutete Ursache als Tatsache

Die Diagnose wird früh auf eine möglicherweise falsche Richtung festgelegt.

Fehler 3: nur technische Komponente erfassen

Betroffener Service und Geschäftsbezug bleiben unbekannt.

Fehler 4: meldende und betroffene Person verwechseln

Kommunikation und Auswirkung werden falsch zugeordnet.

Fehler 5: Kategorie nach Supportteam auswählen

Organisatorische Änderungen machen das Modell unbrauchbar.

Fehler 6: zu viele Kategorieebenen

Erfassung wird langsam und fehleranfällig.

Fehler 7: „Sonstiges“ als Standard verwenden

Trends und Zuständigkeiten bleiben unsichtbar.

Fehler 8: Kategorie nach neuen Erkenntnissen nicht korrigieren

Berichte und Wissenszuordnung bleiben falsch.

Fehler 9: Maßnahmen ohne Ergebnis dokumentieren

Es ist nicht erkennbar, ob ein Test erfolgreich war.

Fehler 10: interne Abkürzungen ohne Erklärung

Andere Teams und spätere Bearbeiter verstehen die Notizen nicht.

Fehler 11: Ticketweiterleitung ohne Kontext

Nächste Gruppe muss die gesamte Situation erneut untersuchen.

Fehler 12: Status „Warten“ ohne nächsten Schritt

Der Vorgang bleibt unbegrenzt liegen.

Fehler 13: Ticket schließen, sobald eine technische Aktion durchgeführt wurde

Das Benutzer-Outcome wurde noch nicht bestätigt.

Fehler 14: Duplikate einfach löschen

Zusätzliche Auswirkungen und Kontaktinformationen gehen verloren.

Fehler 15: sensible Daten unkontrolliert speichern

Datenschutz- und Sicherheitsrisiken entstehen.

Fehler 16: alle Felder verpflichtend machen

Benutzer und Mitarbeiter tragen Platzhalter oder falsche Werte ein.

Fehler 17: KI-Kategorie ungeprüft übernehmen

Kritische oder sicherheitsbezogene Meldungen können falsch eingeordnet werden.

Fehler 18: Kategorien nie überarbeiten

Neue Services und veränderte Arbeitsweisen werden nicht berücksichtigt.

Bedeutung für Fachinformatiker für Systemintegration

Fachinformatiker erhalten häufig Tickets aus dem Service Desk oder erstellen selbst technische Vorgänge.

Vor Beginn der Bearbeitung sollte geprüft werden:

- Welcher Service ist betroffen?
- Was ist das tatsächliche Symptom?

- Welche Auswirkungen bestehen?
- Welche Informationen sind bestätigt?
- Welche Vermutungen wurden nur angenommen?
- Was wurde bereits geprüft?
- Welche Ergebnisse liegen vor?
- Welche Systeme und Abhängigkeiten sind betroffen?
- Wer besitzt den Vorgang?
- Wann erwartet der Benutzer eine Rückmeldung?

Während der Bearbeitung sollten dokumentiert werden:

- technische Prüfungen,
- konkrete Ergebnisse,
- Änderungen,
- Zeitpunkte,
- Risiken,
- Rückfallmaßnahmen,
- und nächste Schritte.

Nach der Bearbeitung sollten dokumentiert werden:

- wiederhergestellte Funktion,
- Benutzerbestätigung,
- technische Ursache, falls bekannt,
- dauerhafte oder vorläufige Lösung,
- offene Folgeaktivitäten,
- und mögliche Verbesserungen.

30-Sekunden-Prüfung eines Tickets

1. **Person:** Wer meldet und wer ist betroffen?
2. **Service:** Welcher Service oder Arbeitsprozess ist betroffen?
3. **Symptom:** Was wurde tatsächlich beobachtet?
4. **Zeit:** Seit wann besteht die Situation?
5. **Auswirkung:** Was kann nicht durchgeführt werden?
6. **Umfang:** Wer oder was ist zusätzlich betroffen?
7. **Einordnung:** Incident, Request, Frage, Feedback oder Sicherheitsmeldung?
8. **Kategorie:** Unterstützt die Kategorie Bearbeitung und Auswertung?
9. **Maßnahmen:** Was wurde bereits durchgeführt?
10. **Ergebnisse:** Was haben die Maßnahmen gezeigt?
11. **Ownership:** Wer ist aktuell verantwortlich?
12. **Nächster Schritt:** Was geschieht als Nächstes?
13. **Kommunikation:** Wann erhält der Benutzer eine Rückmeldung?
14. **Sicherheit:** Enthält das Ticket nur zulässige Informationen?
15. **Abschluss:** Wie wird das tatsächliche Outcome bestätigt?

Checkliste für die Ersterfassung

- [ ] Ist die meldende Person bekannt?
- [ ] Ist die tatsächlich betroffene Person oder Gruppe bekannt?
- [ ] Ist der betroffene Service bestimmt?
- [ ] Ist der Tickettitel eindeutig?
- [ ] Ist das beobachtete Symptom beschrieben?
- [ ] Ist der erwartete Normalzustand bekannt?
- [ ] Ist der Beginn der Beeinträchtigung erfasst?
- [ ] Sind Auswirkungen und Umfang dokumentiert?
- [ ] Ist eine vorhandene Zwischenlösung bekannt?
- [ ] Ist die Vorgangsart vorläufig bestimmt?
- [ ] Ist die Kategorie verständlich und passend?
- [ ] Wurden sensible Daten vermieden?
- [ ] Ist der nächste Schritt festgelegt?
- [ ] Wurde ein realistischer Informationszeitpunkt genannt?

Checkliste für technische Arbeitsnotizen

- [ ] Ist jede Maßnahme mit Zeitpunkt dokumentiert?
- [ ] Ist der Zweck der Maßnahme erkennbar?
- [ ] Ist das Ergebnis der Maßnahme beschrieben?
- [ ] Sind Beobachtung und Vermutung getrennt?
- [ ] Sind technische Werte ausreichend präzise?
- [ ] Werden interne Abkürzungen erklärt?
- [ ] Sind Änderungen und Rückfallmaßnahmen dokumentiert?
- [ ] Wurden Zugangsdaten und sensible Inhalte entfernt?
- [ ] Ist der nächste technische Schritt erkennbar?
- [ ] Kann eine andere Person die Bearbeitung übernehmen?

Checkliste vor einer Weiterleitung

- [ ] Ist die Zielgruppe fachlich geeignet?
- [ ] Ist der betroffene Service eindeutig?
- [ ] Sind Auswirkung und Priorität nachvollziehbar?
- [ ] Sind bisherige Maßnahmen dokumentiert?

- [ ] Sind Ergebnisse und Fehlermeldungen enthalten?
- [ ] Ist klar, welche Unterstützung benötigt wird?
- [ ] Sind relevante Anhänge sicher beigefügt?
- [ ] Ist die weitere Verantwortung geklärt?
- [ ] Ist die Benutzerkommunikation geregelt?
- [ ] Wurde unnötiges Ticket-Pingpong vermieden?

Checkliste für die Kategorisierung

- [ ] Ist die Vorgangsart korrekt?
- [ ] Ist der betroffene Service erfasst?
- [ ] Beschreibt die Kategorie das Anliegen statt nur das Team?
- [ ] Ist die Symptomkategorie angemessen?
- [ ] Wurde keine unbestätigte Ursache als Kategorie verwendet?
- [ ] Kann die Kategorie bei neuen Erkenntnissen geändert werden?
- [ ] Wird „Sonstiges“ nur begründet verwendet?
- [ ] Sind Kategorien für die jeweilige Zielgruppe verständlich?
- [ ] Unterstützt die Kategorie Berichte oder Bearbeitung?
- [ ] Ist die Anzahl der Kategorieebenen angemessen?

Checkliste für Anhänge und sensible Informationen

- [ ] Wird der Anhang tatsächlich benötigt?
- [ ] Sind Kennwörter, Tokens und Schlüssel entfernt?
- [ ] Sind personenbezogene Daten auf das notwendige Maß reduziert?
- [ ] Sind vertrauliche Inhalte geschützt?
- [ ] Besitzen nur berechtigte Personen Zugriff?
- [ ] Ist der Übertragungsweg geeignet?
- [ ] Sind Logdaten ausreichend gekürzt oder anonymisiert?
- [ ] Darf der Anhang an Lieferanten weitergegeben werden?
- [ ] Sind Aufbewahrungs- und Löschregeln berücksichtigt?
- [ ] Muss ein Sicherheitsvorfall eröffnet werden?

Checkliste vor dem Abschluss

- [ ] Ist das ursprüngliche Anliegen erneut geprüft worden?
- [ ] Ist die technische Funktion wiederhergestellt oder der Request erfüllt?
- [ ] Kann der Benutzer den benötigten Arbeitsschritt durchführen?
- [ ] Ist die Lösung oder Erfüllung verständlich dokumentiert?
- [ ] Sind Ursache und Lösung korrekt kategorisiert, soweit bekannt?
- [ ] Sind offene Folgeaufgaben verknüpft?
- [ ] Muss ein Problem Record erstellt werden?
- [ ] Muss ein Knowledge-Artikel erstellt oder aktualisiert werden?
- [ ] Muss eine Verbesserung erfasst werden?
- [ ] Wurde der Benutzer über das Ergebnis informiert?
- [ ] Sind vertrauliche Daten ordnungsgemäß behandelt?
- [ ] Sind die Abschlusskriterien erfüllt?

Checkliste für das Kategoriemodell der Organisation

- [ ] Orientieren sich Benutzerkategorien an verständlichen Services und Anliegen?
- [ ] Sind interne technische Kategorien getrennt pflegbar?
- [ ] Besitzt jede Kategorie einen erkennbaren Nutzen?
- [ ] Werden unnötig tiefe Hierarchien vermieden?
- [ ] Sind Kategorien nicht ausschließlich an Teams gebunden?
- [ ] Können Symptome, Ursachen und Lösungen getrennt erfasst werden?
- [ ] Wird die Kategorie „Sonstiges“ regelmäßig ausgewertet?
- [ ] Werden Fehlzuordnungen gemessen?
- [ ] Werden neue Services und Anforderungen ergänzt?
- [ ] Werden veraltete Kategorien entfernt?
- [ ] Sind Zuweisungsregeln transparent?
- [ ] Werden Benutzer und Bearbeiter bei Änderungen informiert?
- [ ] Fließen Erkenntnisse in Continual Improvement ein?

Schnellreferenz

Bereich	Zentrale Frage
Titel	Ist die Situation auf einen Blick verständlich?
Person	Wer meldet und wer ist betroffen?
Service	Welche Arbeitsfähigkeit oder Leistung ist beeinträchtigt?

Bereich	Zentrale Frage
Symptom	Was wurde tatsächlich beobachtet?
Zeit	Wann begann die Situation?
Auswirkung	Was kann nicht durchgeführt werden?
Vorgangsart	Incident, Request, Frage, Feedback oder Sicherheitsmeldung?
Kategorie	Unterstützt sie Bearbeitung, Wissen und Auswertung?
Maßnahmen	Was wurde bereits getan?
Ergebnisse	Was wurde dadurch festgestellt?
Ownership	Wer besitzt den nächsten Schritt?
Kommunikation	Wann erhält der Benutzer die nächste Information?
Sicherheit	Sind Daten und Anhänge angemessen geschützt?
Abschluss	Wurde das tatsächliche Outcome bestätigt?

Zusammenfassende Darstellung

> Benutzerkontakt oder technische Meldung
> ↓
> meldende und betroffene Personen bestimmen
> ↓
> betroffenen Service und tatsächliches Symptom erfassen
> ↓
> Auswirkungen, Umfang und Zeitangaben dokumentieren
> ↓
> Vorgangsart und vorläufige Kategorie bestimmen
> ↓
> Ownership und nächsten Schritt festlegen
> ↓
> Maßnahmen und Ergebnisse chronologisch dokumentieren
> ↓
> Kategorie bei neuen Erkenntnissen korrigieren
> ↓
> verknüpfte Incidents, Requests, Problems, Changes und Lieferantenfälle sichtbar machen
> ↓
> Lösung oder Erfüllung dokumentieren
> ↓
> Benutzer-Outcome bestätigen

> ↓
> Wissen, Muster und Verbesserungsmöglichkeiten weiterverwenden

---

Verwandte Seiten

- 3.1 Der Service Desk als zentraler Kontaktpunkt
- 3.2 Kontaktkanäle und Erreichbarkeit
- 3.3 Benutzerkommunikation und professioneller Umgang
- 3.5 Auswirkungen, Dringlichkeit und Priorität
- 3.6 Erstdiagnose, Lösung und funktionale Eskalation
- 3.7 Ownership, hierarchische Eskalation und Major Incidents
- 3.8 Self-Service, Wissensnutzung und Automatisierung
- 3.9 Kennzahlen, Qualität und Continual Improvement im Service Desk
- Incident Management
- Service Request Management
- Problem Management
- Knowledge Management
- Service Configuration Management
- Information Security Management
- Measurement and Reporting

Quellen und Versionsstand

Offizielle Grundlagen

- PeopleCert: ITIL 4 Practitioner – Service Desk
- PeopleCert: ITIL 4 Practitioner – Incident Management
- PeopleCert: ITIL 4 Practitioner – Service Request Management
- PeopleCert: ITIL 4 Specialist – Monitor, Support and Fulfil
- PeopleCert: ITIL 4 Management Practices 2023
- PeopleCert ATV: Essential Features to Look for in ITSM Software
- PeopleCert ATV: ITSM Software for Incident Management
- ITIL: ITIL Service – Version 5
- ITIL: ITIL Service Version 5 – An Enterprise-Wide Approach to Service Thinking
- ITIL: ITIL Foundation Version 5 – What’s New?

Offiziell bestätigter Stand

Die offiziellen ITIL- und PeopleCert-Quellen bestätigen:

- Service Desk, Incident Management und Service Request Management bleiben zentrale ITIL Management Practices.
- Die Practices umfassen mehr als einzelne Prozesse und berücksichtigen Rollen, Informationen, Technologien, Wissen, Kommunikation und Wertströme.
- Incident-Management-Werkzeuge unterstützen typischerweise die Erfassung, Kategorisierung, Priorisierung, Verlaufsdokumentation, Eskalation und Statusverfolgung.
- Service-Desk-Prozesse und Informationen sollen in die Wertströme der Organisation integriert werden.
- Datenqualität, Kommunikation, Nachvollziehbarkeit und geeignete Werkzeugunterstützung beeinflussen Serviceleistung und Benutzererfahrung.
- ITIL Version 5 führt die Management Practices fort und richtet digitale Produkt- und Servicearbeit stärker an Wert, Outcomes, Erfahrung und Ende-zu-Ende-Zusammenarbeit aus.

Einordnung

Die auf dieser Seite dargestellten:

- Ticketfelder,
- Kategoriemodelle,
- Statusbezeichnungen,
- Titelregeln,
- Checklisten,
- Pflichtfeldempfehlungen,
- Beispiele,
- und Qualitätskriterien

sind herstellerneutrale redaktionelle Praxisempfehlungen dieses unabhängigen Nachschlagewerks.

ITIL schreibt kein universelles:

- Ticketformular,
- Feldschema,
- Kategoriemodell,
- Statusmodell,
- Supportgruppenmodell,
- Prioritätsmodell,
- oder Ticketsystem

für alle Organisationen vor.

Die konkrete Gestaltung muss angepasst werden an:

- Services,
- Benutzergruppen,
- Wertströme,
- Risiken,

- Sicherheits- und Datenschutzanforderungen,
- Lieferanten,
- Fähigkeiten,
- und technische Werkzeuge.

Behandelter Framework-Stand: ITIL Version 5
Zusätzlich berücksichtigt: aktuelle öffentlich zugängliche ITIL-4-Practice-Guidance
Fachlicher Stand: August 2026
