3.7 Ownership, hierarchische Eskalation und Major Incidents (Teil 1/2)

> **Kurz erklärt**
>
> Jeder Incident muss jederzeit einen eindeutig verantwortlichen Bearbeiter oder eine verantwortliche Gruppe besitzen.
>
> ITIL bezeichnet dies als **Ownership**.
>
> Ownership bedeutet nicht zwingend, dass diese Person den Fehler selbst behebt. Sie trägt jedoch die Verantwortung dafür, dass der Incident aktiv verfolgt, koordiniert und bis zum Abschluss begleitet wird.

Was bedeutet Ownership?

Ownership beantwortet die Frage:

> **Wer kümmert sich aktuell um diesen Incident?**

Der Owner sorgt dafür, dass:

- der Incident aktiv bearbeitet wird,
- keine unnötigen Wartezeiten entstehen,
- Eskalationen rechtzeitig erfolgen,
- Benutzer informiert werden,
- der nächste Schritt bekannt ist,
- und der Vorgang sauber abgeschlossen wird.

Ownership bleibt bestehen – auch wenn andere Teams beteiligt werden.

Ownership ist nicht gleich Bearbeitung

Ein häufiger Irrtum:

> Wer gerade technisch arbeitet, ist automatisch für den gesamten Incident verantwortlich.

Das stimmt nicht.

Beispiel:

Der Service Desk erstellt ein Ticket und eskaliert es an das Netzwerkteam.

Während das Netzwerkteam analysiert, kann der Service Desk weiterhin Owner bleiben und:

- Rückfragen koordinieren,
- den Benutzer informieren,
- Fortschritte überwachen,
- Zielzeiten prüfen,
- und notwendige Eskalationen auslösen.

Der technische Bearbeiter führt also eine konkrete Aufgabe aus.

Der Owner sorgt dafür, dass der Incident insgesamt gesteuert wird.

Zuständigkeit, Bearbeitung und Ownership unterscheiden

Begriff	Bedeutung
Ownership	Verantwortung für den gesamten Incident bis zum Abschluss
Bearbeitung	Durchführung aktueller technischer oder organisatorischer Maßnahmen
Zuständigkeit	fachliche Verantwortung für einen bestimmten Service, Bereich oder eine Komponente
Kommunikationsverantwortung	Verantwortung für Statusmeldungen an Benutzer und Stakeholder
Entscheidungsverantwortung	Befugnis, Prioritäten, Ressourcen oder Vorgehen festzulegen

Mehrere Personen oder Teams können gleichzeitig an einem Incident arbeiten.

Der Owner sollte jedoch eindeutig festgelegt sein.

Warum Ownership wichtig ist

Ohne klaren Owner entstehen häufig:

- doppelte Arbeiten,
- lange Wartezeiten,
- vergessene Tickets,
- fehlende Kommunikation,

- unklare Verantwortlichkeiten,
- Ticket-Pingpong,
- verspätete Eskalationen,
- und unklare Abschlüsse.

Typische Aussage:

> „Ich dachte, das andere Team kümmert sich darum.

Genau solche Situationen soll Ownership verhindern.

Aufgaben des Owners

Der Owner überwacht den gesamten Incident.

Typische Aufgaben:

- Ticket kontrollieren,
- Bearbeitung verfolgen,
- Priorität prüfen,
- Eskalationen auslösen,
- Benutzer informieren,
- Rückmeldungen koordinieren,
- beteiligte Teams abstimmen,
- Zielzeiten überwachen,
- Workaround oder Lösung nachverfolgen,
- Abschluss prüfen.

Der Owner muss nicht jede technische Aufgabe selbst durchführen.

Er muss aber sicherstellen, dass der Incident nicht stehen bleibt.

Ownership während einer funktionalen Eskalation

Auch nach einer funktionalen Eskalation bleibt Ownership bestehen.

Beispiel:

```
Benutzer
   ↓
Service Desk (Owner)
   ↓
```

```
Netzwerkteam
   ↓
Hersteller
```

Das Netzwerkteam untersucht den Fehler.

Der Service Desk sorgt weiterhin dafür, dass:

- Statusmeldungen erfolgen,
- der Benutzer informiert wird,
- Fristen eingehalten werden,
- offene Rückfragen geklärt werden,
- und der Vorgang bis zur Wiederherstellung verfolgt wird.

Eine Eskalation bedeutet also nicht automatisch, dass die Gesamtverantwortung verschwindet.

Ownership übertragen

In manchen Organisationen kann Ownership wechseln.

Beispiele:

- Service Desk → Netzwerkteam
- Netzwerkteam → Incident Manager
- Incident Manager → Major Incident Manager
- Service Desk → Produktteam
- internes Team → externer Provider mit interner Koordination

Dabei sollte immer dokumentiert werden:

- Zeitpunkt,
- bisheriger Owner,
- neuer Owner,
- Grund der Übergabe,
- aktueller Status,
- nächster Schritt,
- und Kommunikationsverantwortung.

Eine Übergabe ist nur dann sinnvoll, wenn die neue Verantwortung eindeutig akzeptiert wurde.

Ein Ticket darf niemals besitzerlos sein

Ein Incident ohne Owner ist eines der größten Risiken im Support.

Jeder Incident sollte jederzeit erkennen lassen:

- Wer ist verantwortlich?
- Wer arbeitet aktuell?
- Wer informiert den Benutzer?
- Wer entscheidet über Eskalationen?
- Wer verfolgt Zielzeiten?
- Wer prüft die Wiederherstellung?
- Wer schließt den Vorgang ab?

> **Merke**
>
> Ein zugewiesenes Ticket ist nicht automatisch ein aktiv gesteuerter Incident.

Funktionale und hierarchische Eskalation unterscheiden

Diese beiden Begriffe werden häufig verwechselt.

Funktionale Eskalation	Hierarchische Eskalation
fehlendes Fachwissen oder fehlende Berechtigung	Management-Unterstützung oder Entscheidung erforderlich
Spezialisten notwendig	Priorisierung, Ressourcen oder Risikoentscheidung notwendig
technischer Fokus	organisatorischer oder geschäftlicher Fokus
Beispiel: Netzwerkteam einbinden	Beispiel: Service Owner oder Management informieren
Ziel: richtige Fähigkeit einbinden	Ziel: Hindernisse, Risiken oder Entscheidungen klären

---

Funktionale Eskalation

Die funktionale Eskalation wurde bereits in **3.6 Erstdiagnose, Lösung und funktionale Eskalation** behandelt.

Beispiele:

- Service Desk → Netzwerkteam
- Service Desk → Serverteam
- Service Desk → Datenbankteam
- Service Desk → Cloud-Team
- internes Team → Hersteller oder Lieferant

Ziel:

Die richtigen Spezialisten bearbeiten den Incident.

Eine funktionale Eskalation ist keine Schuldzuweisung.

Sie bedeutet nur, dass zusätzliche Fähigkeiten, Werkzeuge oder Berechtigungen benötigt werden.

Hierarchische Eskalation

Bei einer hierarchischen Eskalation wird nicht primär weiteres Fachwissen benötigt.

Benötigt werden beispielsweise:

- Entscheidungen,
- Ressourcen,
- Priorisierung,
- Budget,
- Management-Unterstützung,
- Freigaben,
- Konfliktlösung,
- oder Kommunikation auf höherer Ebene.

Beispiele:

- SLA wird voraussichtlich verletzt.
- Mehrere Fachbereiche sind betroffen.
- Management muss informiert werden.
- Ein externer Kunde eskaliert.
- Ein Lieferant reagiert nicht.
- Zusätzliche Ressourcen werden benötigt.
- Ein Risiko kann nicht auf operativer Ebene entschieden werden.
- Erheblicher Imageschaden ist möglich.

Wann sollte eskaliert werden?

Nicht erst, wenn:

- ein SLA bereits verletzt wurde,
- der Benutzer mehrfach angerufen hat,
- Management sich beschwert,
- oder ein Incident längere Zeit unbeachtet blieb.

Sondern bereits dann, wenn erkennbar wird:

- ein Ziel wird wahrscheinlich verfehlt,
- weitere Unterstützung wird benötigt,
- Risiken steigen,
- Ressourcen nicht ausreichen,

- Entscheidungen fehlen,
- die Kommunikation nicht mehr ausreicht,
- oder der Incident eine besondere geschäftliche Bedeutung bekommt.

> **Merke**
>
> Gute Eskalation ist rechtzeitig, begründet und handlungsorientiert.

Eskalation bedeutet keine Schuld

Eine Eskalation ist keine Kritik an einzelnen Mitarbeitern.

Sie dient dazu:

- Hindernisse zu beseitigen,
- Entscheidungen schneller zu treffen,
- zusätzliche Ressourcen bereitzustellen,
- Risiken sichtbar zu machen,
- Kommunikation zu verbessern,
- und den Service schneller wiederherzustellen.

Ungeeignet:

> Wir eskalieren, weil das Team es nicht schafft.

Besser:

> Wir eskalieren, weil die Zielzeit gefährdet ist und zusätzliche Entscheidungskompetenz benötigt wird.

Kommunikationseskalation

Manchmal funktioniert die technische Bearbeitung, aber die Kommunikation reicht nicht aus.

Beispiele:

- Benutzer erhält keine Rückmeldung.
- Management verlangt Statusberichte.
- Mehrere Standorte müssen informiert werden.

- Es gibt widersprüchliche Aussagen.
- Ein Kunde benötigt eine abgestimmte externe Meldung.
- Der Service Desk besitzt keine freigegebene Kommunikationslinie.

Auch hierfür kann eine Eskalation notwendig sein.

Die technische Lösung und die Kommunikation müssen beide gesteuert werden.

Major Incident

Ein **Major Incident** ist ein besonders schwerwiegender Incident.

Er besitzt normalerweise:

- hohe Auswirkungen,
- hohe Dringlichkeit,
- viele betroffene Benutzer,
- kritische betroffene Services,
- erhebliche geschäftliche Folgen,
- erhöhten Kommunikationsbedarf,
- oder besondere Risiken.

ITIL schreibt keine universelle feste Definition vor.

Jede Organisation muss eigene Kriterien festlegen.

Mögliche Kriterien für einen Major Incident

Beispiele:

- gesamte Produktion steht,
- mehrere Standorte sind betroffen,
- ein geschäftskritischer Service ist ausgefallen,
- zentrale Anmeldung funktioniert nicht,
- kritische Cloud-Dienste sind nicht verfügbar,
- ein Sicherheitsvorfall wird vermutet,
- externer Kundenimpact ist erheblich,
- keine geeignete Zwischenlösung ist verfügbar,
- gesetzliche oder vertragliche Fristen sind gefährdet,
- Management- oder Krisenkommunikation ist notwendig.

Die Kriterien sollten dokumentiert und regelmäßig überprüft werden.

Major Incident ist nicht automatisch P1

Viele Organisationen setzen:

> P1 = Major Incident

Dies ist jedoch keine allgemeine ITIL-Vorgabe.

Ein P1-Incident kann ein Major Incident sein.

Er muss es aber nicht in jeder Organisation automatisch sein.

P1	Major Incident
beschreibt häufig die höchste operative Priorität	beschreibt einen besonders schwerwiegenden Incident mit besonderer Koordination
steuert Zielzeiten und Bearbeitungsreihenfolge	aktiviert zusätzliche Rollen, Kommunikation und Steuerung
kann durch normale Supportstruktur bearbeitet werden	benötigt häufig besondere Koordination
wird meist aus Auswirkung und Dringlichkeit bestimmt	kann zusätzliche Kriterien enthalten

> **Merke**
>
> Priorität und Major-Incident-Status sollten bewusst unterschieden oder eindeutig gemeinsam definiert werden.

Ziele eines Major-Incident-Verfahrens

Bei normalen Incidents liegt der Fokus häufig auf der technischen Wiederherstellung.

Bei Major Incidents kommen zusätzliche Aufgaben hinzu:

- Koordination mehrerer Teams,
- schnelle Entscheidungsfindung,
- klare Kommunikation,
- Priorisierung von Ressourcen,
- Management-Unterstützung,
- Lieferantensteuerung,
- Risikoüberwachung,
- Dokumentation unter Zeitdruck,
- und spätere Nachbereitung.

Das Ziel bleibt die Wiederherstellung des Services.

Der Weg dorthin benötigt jedoch mehr Steuerung.

Major Incident Manager

Viele Organisationen benennen für Major Incidents einen eigenen **Major Incident Manager**.

Diese Person repariert normalerweise nichts selbst.

Typische Aufgaben:

- Teams koordinieren,
- Status sammeln,
- Entscheidungen vorbereiten,
- Kommunikationsfreigaben abstimmen,
- Prioritäten überwachen,
- Eskalationen auslösen,
- War Room steuern,
- Statusintervalle einhalten,
- und die Nachbereitung vorbereiten.

Der Major Incident Manager sorgt dafür, dass Spezialisten arbeiten können, ohne die Gesamtkoordination nebenbei leisten zu müssen.

Typischer Ablauf eines Major Incidents

```
Incident erkannt
      ↓
erste Bewertung
      ↓
Major-Incident-Kriterien prüfen
      ↓
Major Incident bestätigen oder ablehnen
      ↓
Owner und Major Incident Manager festlegen
      ↓
Fachteams und Lieferanten einbinden
      ↓
Kommunikation starten
      ↓
Workaround oder Wiederherstellung umsetzen
      ↓
```

```
Service stabilisieren
        ↓
Abschluss und Nachbereitung
```

Nicht jede Organisation verwendet exakt diesen Ablauf.

Entscheidend ist eine klare Koordination.

Sofortmaßnahmen bei Major Incidents

Zu Beginn sollte schnell geklärt werden:

- Was ist betroffen?
- Welche Services sind betroffen?
- Welche Standorte sind betroffen?
- Welche Benutzer oder Kunden sind betroffen?
- Welche geschäftlichen Auswirkungen bestehen?
- Seit wann besteht die Störung?
- Gibt es einen sicheren Workaround?
- Welche Teams werden benötigt?
- Gibt es Hinweise auf einen Sicherheitsvorfall?
- Wer kommuniziert offiziell?
- Wann erfolgt die nächste Statusmeldung?

Erst danach beginnt die tiefere technische Analyse.

Rollen bei Major Incidents

Typische Rollen können sein:

- Service Desk,
- Incident Manager,
- Major Incident Manager,
- technische Spezialisten,
- Service Owner,
- Kommunikationsverantwortliche,
- Management,
- Informationssicherheit,
- Lieferanten,
- Fachbereich,
- Problem Manager.

Nicht jede Organisation benötigt alle Rollen.

Wichtig ist, dass Zuständigkeiten und Entscheidungswege bekannt sind.

War Room

Bei kritischen Incidents wird häufig ein gemeinsamer Kommunikationskanal eingerichtet.

Beispiele:

- Telefonkonferenz,
- Videokonferenz,
- Microsoft Teams,
- Slack,
- spezieller Incident-Kanal,
- Statusboard.

Vorteile:

- schnellere Abstimmung,
- weniger Rückfragen,
- keine parallelen Einzelgespräche,
- gemeinsame Faktenlage,
- bessere Koordination.

Ein War Room benötigt klare Regeln.

Sonst entsteht schnell Unübersichtlichkeit.

Regeln im War Room

Sinnvolle Regeln:

- eine Person koordiniert,
- Fakten werden von Vermutungen getrennt,
- Maßnahmen werden vor Ausführung abgestimmt,
- Ergebnisse werden dokumentiert,
- Kommunikation nach außen erfolgt abgestimmt,
- Nebenunterhaltungen werden begrenzt,
- Entscheidungen werden festgehalten,
- nächste Schritte werden eindeutig zugewiesen.

Ungeeignet:

> “ Alle diskutieren gleichzeitig technische Vermutungen ohne Dokumentation.

Besser:

> Hypothesen werden gesammelt, priorisiert, getestet und mit Ergebnis dokumentiert.

Kommunikation wird wichtiger

Je größer der Incident, desto wichtiger wird regelmäßige Kommunikation.

Stakeholder möchten wissen:

- Was ist passiert?
- Welche Auswirkungen bestehen?
- Was wird aktuell getan?
- Gibt es einen Workaround?
- Was sollen Benutzer tun oder nicht tun?
- Wann gibt es neue Informationen?

Auch wenn noch keine Lösung vorliegt, ist eine Statusmeldung sinnvoll.

Beispiel:

> Die Ursache ist noch nicht bestätigt. Netzwerk- und Identitätsdienste werden geprüft. Ein sicherer Workaround ist derzeit nicht verfügbar. Die nächste Statusmeldung erfolgt um 11:30 Uhr.

Statusintervalle

Für kritische Incidents sollten Statusintervalle festgelegt werden.

Beispiel:

- alle 30 Minuten bei Major Incident,
- alle 60 Minuten bei hoher Priorität,
- nach wesentlichen Änderungen bei normaler Priorität.

Die konkreten Zeiten legt die Organisation selbst fest.

Nicht geeignet:

>

> Wir melden uns, wenn alles wieder funktioniert.

Besser:

> „Die nächste Statusmeldung erfolgt spätestens um 10:30 Uhr, auch wenn bis dahin noch keine vollständige Lösung vorliegt.

Single Point of Communication

Bei großen Incidents sollte möglichst eine abgestimmte Kommunikationsquelle verwendet werden.

Beispiele:

- Incident Manager,
- Kommunikationsverantwortlicher,
- Statusseite,
- Service Desk mit freigegebenem Text,
- zentrale E-Mail oder Portalnachricht.

Dadurch werden widersprüchliche Aussagen vermieden.

Technische Detaildiskussionen gehören nicht ungefiltert in Benutzerkommunikation.

Stakeholder zielgruppengerecht informieren

Zielgruppe	Benötigte Informationen
Benutzer	Auswirkungen, Workaround, nächste Statusmeldung
Service Desk	freigegebene Formulierung, bekannte Symptome, Ticketverknüpfung
technische Teams	Logs, Hypothesen, Messergebnisse, Maßnahmen
Management	geschäftliche Auswirkungen, Risiken, erwartete Entscheidungen
Kunden	Serviceverfügbarkeit, Einschränkungen, nächstes Update
Lieferanten	technische Nachweise, Versionen, Priorität, gewünschte Unterstützung
Informationssicherheit	Sicherheitsbezug, Beweise, Risiko, Schutzmaßnahmen

Nicht jeder Empfänger benötigt dieselbe Detailtiefe.

Ressourcen koordinieren

Bei Major Incidents arbeiten oft mehrere Teams gleichzeitig.

Der Major Incident Manager oder Incident Coordinator achtet darauf, dass:

- Aufgaben verteilt werden,
- Doppelarbeit vermieden wird,
- Ergebnisse zusammengeführt werden,
- blockierende Entscheidungen eskaliert werden,
- Lieferanten eingebunden werden,
- und niemand ohne Abstimmung riskante Änderungen durchführt.

Dokumentation während eines Major Incidents

Auch unter Zeitdruck muss dokumentiert werden.

Mindestens festhalten:

- Zeitpunkt,
- Beobachtung,
- Maßnahme,
- Ergebnis,
- Entscheidung,
- Verantwortlicher,
- nächster Schritt,
- Kommunikationszeitpunkt.

Diese Informationen werden später benötigt für:

- Review,
- Problem Management,
- Audits,
- Lessons Learned,
- und Continual Improvement.

Nach der Wiederherstellung

Mit der technischen Wiederherstellung endet die Arbeit häufig noch nicht.

Es folgen möglicherweise:

- Stabilitätsüberwachung,
- Benutzerinformation,
- Abschlusskommunikation,

- Ursachenanalyse,
- Problem Record,
- Review,
- Verbesserung der Dokumentation,
- Monitoring-Anpassung,
- Aktualisierung von Runbooks,
- und Bewertung der Kommunikation.

Major Incident Review

Nach Abschluss sollte geprüft werden:

- Was ist passiert?
- Wann begann die Störung?
- Wann wurde sie erkannt?
- Wann wurde eskaliert?
- Welche Maßnahmen waren wirksam?
- Wo gab es Verzögerungen?
- War die Kommunikation ausreichend?
- Waren Rollen und Verantwortlichkeiten klar?
- Hat der Workaround funktioniert?
- Welche Verbesserungen sind notwendig?

Ziel ist Lernen, nicht Schuldzuweisung.

Lessons Learned

Typische Ergebnisse können sein:

- bessere Dokumentation,
- neue Checklisten,
- Monitoring erweitern,
- Alarmgrenzen anpassen,
- Runbooks verbessern,
- Schulungen durchführen,
- Automatisierungen ergänzen,
- Serviceabhängigkeiten in der CMDB korrigieren,
- Lieferanteneskalation verbessern,
- Kommunikationsvorlagen überarbeiten.

Lessons Learned haben nur dann Wert, wenn daraus konkrete Maßnahmen entstehen.

Problem Management einbeziehen

Ein Major Incident führt häufig zu einem Problem Record.

Dadurch kann später untersucht werden:

- eigentliche Ursache,
- beitragende Faktoren,
- bekannte Fehler,
- Workaround,
- dauerhafte Lösung,
- notwendiger Change,
- Verbesserungsmaßnahmen.

Incident Management stellt den Service wieder her.

Problem Management sorgt dafür, dass die Ursache verstanden und zukünftige Wiederholungen möglichst vermieden werden.

Typische Fehler

Fehler 1: Kein eindeutiger Owner

Niemand verfolgt den Incident Ende zu Ende.

Fehler 2: Ownership geht bei Eskalation verloren

Das Ticket wird weitergegeben, aber niemand koordiniert mehr.

Fehler 3: Zu späte Eskalation

Erst nach SLA-Verletzung oder Managementbeschwerde wird reagiert.

Fehler 4: Funktionale und hierarchische Eskalation werden verwechselt

Es wird ein Spezialist benötigt, aber Management informiert – oder umgekehrt.

Fehler 5: Major Incident wird nicht aktiviert

Kriterien sind unklar oder niemand entscheidet.

Fehler 6: Jeder kommuniziert etwas anderes

Es fehlt ein Single Point of Communication.

Fehler 7: Technische Vermutungen werden als Fakten veröffentlicht

Vertrauen geht verloren, wenn Aussagen später korrigiert werden müssen.

Fehler 8: Keine Dokumentation während der Störung

Nachbereitung und Ursachenanalyse werden erschwert.

Fehler 9: War Room ohne Koordination

Viele Personen diskutieren, aber wenige Maßnahmen werden entschieden.

Fehler 10: Nach Wiederherstellung keine Nachbereitung

Die Organisation lernt nicht aus dem Incident.

Fehler 11: Lessons Learned bleiben unverbindlich

Verbesserungen werden beschlossen, aber nicht umgesetzt.

Fehler 12: Problem Management wird nicht gestartet

Die eigentliche Ursache bleibt unbekannt oder unbehandelt.

Praxisbeispiel

08:15 Uhr

Mehrere Standorte melden Ausfälle der zentralen Anmeldung.

08:20 Uhr

Incident wird als Major Incident bewertet.

08:25 Uhr

Major Incident Manager übernimmt die Koordination.

08:30 Uhr

Erste Statusmeldung wird veröffentlicht.

08:40 Uhr

Identity-, Netzwerk- und Cloud-Team arbeiten gemeinsam im War Room.

09:05 Uhr

Ein sicherer Workaround ist verfügbar.

09:40 Uhr

Service ist technisch wiederhergestellt.

10:00 Uhr

Abschlussmeldung wird veröffentlicht.

10:30 Uhr

Review-Termin und Problem Record werden erstellt.

Checkliste Ownership

- [ ] Owner eindeutig festgelegt
- [ ] Bearbeiter und Owner unterschieden
- [ ] Kommunikationsverantwortung geklärt
- [ ] Zielzeiten werden überwacht
- [ ] nächste Schritte sind dokumentiert
- [ ] Eskalationen werden verfolgt
- [ ] Abschluss wird geprüft

Checkliste Eskalation

- [ ] Eskalationsart korrekt gewählt
- [ ] funktionale Eskalation bei fehlendem Fachwissen
- [ ] hierarchische Eskalation bei Entscheidungs- oder Ressourcenbedarf
- [ ] relevante Informationen vollständig
- [ ] richtige Ansprechpartner beteiligt
- [ ] Grund der Eskalation dokumentiert
- [ ] Benutzerkommunikation geregelt
- [ ] Ownership bleibt klar

Checkliste Major Incident

- [ ] Major-Incident-Kriterien geprüft
- [ ] Major Incident bestätigt oder abgelehnt
- [ ] Owner festgelegt
- [ ] Major Incident Manager benannt
- [ ] War Room oder gemeinsamer Kanal eingerichtet

- [ ] beteiligte Teams informiert
- [ ] Lieferanten bei Bedarf eingebunden
- [ ] Kommunikationsverantwortung festgelegt
- [ ] erste Statusmeldung veröffentlicht
- [ ] Statusintervalle festgelegt
- [ ] Workaround geprüft
- [ ] Maßnahmen dokumentiert
- [ ] Wiederherstellung bestätigt
- [ ] Abschlusskommunikation erfolgt
- [ ] Review geplant
- [ ] Problem Record erstellt, falls erforderlich

Bedeutung für Fachinformatiker für Systemintegration

Auch wenn Fachinformatiker nicht immer selbst Incident Manager sind, sind sie häufig maßgeblich an der technischen Analyse und Wiederherstellung beteiligt.

Im Arbeitsalltag bedeutet das:

- strukturiert arbeiten,
- Ergebnisse sauber dokumentieren,
- technische Vermutungen von Fakten trennen,
- Änderungen nachvollziehbar durchführen,
- rechtzeitig eskalieren,
- andere Teams aktiv unterstützen,
- und verständlich kommunizieren.

Gerade bei größeren Störungen entscheidet oft nicht nur technisches Wissen, sondern auch Zusammenarbeit, Ownership und Koordination über die Dauer des Ausfalls.

Zusammenfassung

> Incident übernehmen
> ↓
> Owner eindeutig festlegen
> ↓
> Bearbeitung koordinieren
> ↓
> falls notwendig funktional oder hierarchisch eskalieren

> ↓
> Major Incident aktivieren, wenn Kriterien erfüllt sind
> ↓
> Rollen, Kommunikation und Statusintervalle festlegen
> ↓
> Teams und Lieferanten koordinieren
> ↓
> Service wiederherstellen
> ↓
> Ergebnis stabilisieren und kommunizieren
> ↓
> Review durchführen
> ↓
> Erkenntnisse in Problem Management, Knowledge Management und Continual Improvement übernehmen

---

Verwandte Seiten

- 3.5 Auswirkungen, Dringlichkeit und Priorität
- 3.6 Erstdiagnose, Lösung und funktionale Eskalation
- 3.8 Self-Service, Wissensnutzung und Automatisierung
- Problem Management
- Knowledge Management
- Continual Improvement
- Service Level Management
- Supplier Management

Quellen und Versionsstand

Offizielle Grundlagen

- PeopleCert – ITIL Practice Guide: Incident Management
- PeopleCert – ITIL Practice Guide: Service Desk
- PeopleCert – ITIL Practice Guide: Problem Management
- PeopleCert – ITIL Practice Guide: Service Level Management
- ITIL Foundation – Version 5

Einordnung

Die dargestellten:

- Rollen,
- Eskalationswege,

- Major-Incident-Kriterien,
- Statusintervalle,
- War-Room-Regeln,
- Checklisten,
- und Praxisbeispiele

sind herstellerneutrale Praxisempfehlungen.

ITIL schreibt keine universelle:

- Major-Incident-Definition,
- Eskalationsmatrix,
- Supportstufenstruktur,
- Statusintervallvorgabe,
- War-Room-Struktur,
- oder konkrete Rollenbesetzung

für alle Organisationen vor.

Die konkrete Ausgestaltung muss an:

- Services,
- Risiken,
- Service Levels,
- Organisation,
- Lieferanten,
- Kommunikationswege,
- Sicherheitsanforderungen,
- und Entscheidungsbefugnisse

angepasst werden.

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