Sven Evers

»Alles unter Kontrolle mit Server-Monitoring

Viel Mühe, Zeit und Geld wurde in die Optimierung der Website gesteckt. Das Layout und die Usability sind perfekt, die Ladezeiten rasant, das Suchmaschinen-Ranking im grünen Bereich, Traffic und der Umsatz des Online-Shops wachsen. Alles könnte so schön sein – wenn nicht immer wieder kleinere und größere technische Probleme aufträten und die Erreichbarkeit oder Funktion der Site beeinträchtigten. Experte Sven Evers zeigt am Beispiel der Open-Source-Entwicklung Nagios, wie man das Monitoring von Servern in den Griff bekommt.

Eine typische Situation

Diese Situation kennt jeder Betreiber einer Website: „Irgendetwas" funktioniert nicht – natürlich ausgerechnet nachts, am Wochenende oder wenn durch eine Werbemaßnahme erhöhter Traffic auf der Site herrscht. Die Website läuft vielleicht noch, aber der Mailserver verschickt keine Mails mehr, die Datenbank ist abgestürzt oder eine Tabelle korrupt, die Festplatte vollgelaufen, der FTP-Zugang antwortet nicht mehr, die Anbindung zum Warenwirtschaftssystem ist zusammengebrochen und das sicherheitskritische Update ist auch noch nicht eingespielt. Dann stellt sich noch die Frage: Reicht die Server-Kapazität eigentlich für das anstehende Weihnachtsgeschäft?

Daraus kann schnell ein Horror-Szenario entstehen: der komplette Ausfall der Site, Imageverlust, Umsatzausfall, im Falle einer unbemerkten Sicherheitslücke eventuell sogar rechtliche Konsequenzen.

Schlimm genug, wenn ein Problem auftrifft, das sofort entdeckt wird. Dramatischer können jedoch die Ausfälle sein, die nicht oder erst mit Verzögerung bemerkt werden.

Die Ausgangslage

Server-Software und -Hardware sind robust, keine Frage. Doch schon kleine Shops und etwas anspruchsvollere Websites sind inzwischen sehr komplexe Systeme, viele Bausteine hängen voneinander ab, müssen funktionieren und zusammenspielen. Bestehen erhöhte Ansprüche an Ausfallsicherheit und Performance, reicht ein einzelner Server schon nicht mehr aus und es kommen weitere Hardware-Komponenten wie etwa ein Load-Balancer hinzu. Damit steigt die Wahrscheinlichkeit eines Hardware-Defekts, ebenso die Anforderungen an die verwendete Software.

Automatisierte Angriffswellen schaffen es immer wieder, bekannte und eigentlich schon gestopfte Sicherheitslücken auszunutzen und eine enorme Anzahl von Servern zu kompromittieren – ein deutliches Indiz dafür, dass die meisten Webserver bestenfalls rudimentär gewartet werden.

Für gewissenhafte Systemadministratoren ist der Aufwand enorm, die komplette relevante und (sicherheits-) kritische IT-Infrastruktur mit all ihren Facetten im Auge zu behalten.

Spätestens, wenn bereits der teilweise Ausfall einer Website nicht mehr nur ärgerlich ist, sondern die Geschäftsgrundlage eines Unternehmens gefährdet, ist der Einsatz einer automatisierten Lösung zur Funktionsüberwachung angeraten. Diese Lösung wird mit dem Begriff „Monitoring" beschrieben.

Monitoring – die einfache Lösung

Verschiedene Dienstleister bieten sogenanntes Website-Monitoring an. Dieses überwacht von externen, global verteilten Servern die Erreichbarkeit und Performance, sprich Antwortgeschwindigkeit, einer Website. Je nach Angebot kann auch eine Überprüfung der ausgelieferten Daten erfolgen, also ob tatsächlich die erwarteten Inhalte, eine Fehlermeldung oder nur eine weiße Seite erscheinen. Extern zugängliche Server-Dienste wie z. B. FTP lassen sich ebenfalls

DER AUTOR

Sven Evers ist Geschäftsführer der Evers & Marthaler GmbH und beschäftigt sich seit 1996 mit der Entwicklung und dem Betrieb von Online-Kommunikations- und Vertriebslösungen.

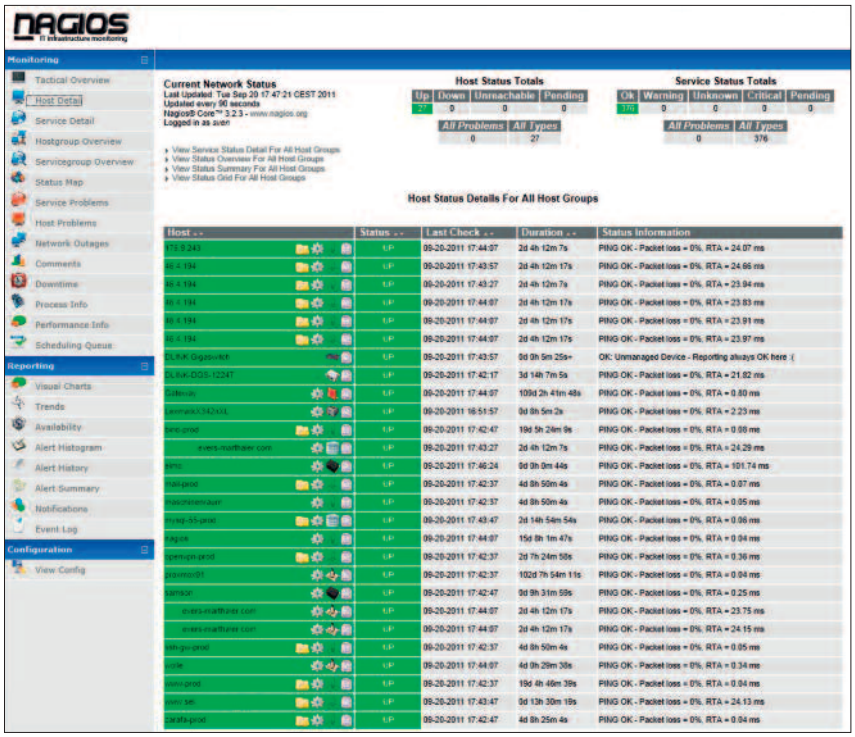

Abbildung 1: Beispiel: Die Nagios-Administrationsoberfläche ist nicht sexy, aber zweckmäßig. Ein Blick genügt – hier ist alles im grünen Bereich.

in ihrer Funktion prüfen. Website-Monitoring-Dienste sind fraglos ein brauchbares Werkzeug und liefern für die Bewertung der weltweiten Performance einer Site wertvolle Daten. Doch sie stoßen schnell an ihre Grenzen und interne Prozesse sind für solche Dienste nicht einzusehen.

Monitoring – die umfassende Lösung

Möchte man wirklich alles, was für den Betrieb einer Website notwendig ist, überwachen, gibt es zum Einsatz einer eigenen, dedizierten IT-Monitoring-Software keine Alternative. Für diesen Zweck existieren verschiedene freie und kommerzielle Software-Pakete. Als Quasi-Standard hat sich die Open-Source-Entwicklung Nagios etabliert, die hier kurz und mit dem Fokus auf Web-Infrastruktur vorgestellt werden soll. Andere Monitoring-Software funktioniert nach ähnlichen Prinzipien.

Der Charme von Nagios liegt vor allem in seiner Flexibilität, seiner Erweiterbarkeit durch Plug-ins, der einfachen Anpassungsmöglichkeit sowie dem niedrigen Resourcenverbrauch. Es existieren Plug-ins für die gängigsten Dienste, etwa Apache und MySQL. Mit ein wenig Programmierkenntnis können eigene Plug-ins für spezielle Aufgaben erstellt werden.

Wie funktioniert das?

Nagios läuft als Dienst auf einem separaten Webserver und prüft von dort in definierbaren Intervallen andere Server und deren Dienste. Zu diesem Zweck „spricht“ Nagios auf Protokollebene (SNMP, TCP, ...) oder mithilfe speziell installierter Clients mit den überwachten Komponenten.

Das kann eine installierte Software (etwa ein Apache), eine interne Eigenschaft (z. B. die CPU-Auslastung) oder eine andere messbare Größe (beispielsweise die Temperatur im Rechnergehäuse) sein.

Die abgefragten Dienste antworten mit einer Statusmeldung (läuft/läuft nicht) oder ggf. einem Wert (Festplatte zu 60 % belegt).

Manche Dienste geben jedoch keine Statusmeldung, z. B. das Amazon Merchant Transport Utility. Hier hilft ein kleiner Umweg. Im genannten Fall lässt sich das Logfile analysieren: Gibt es in einem definierten Zeitraum keine neuen Einträge, ist irgendwo der Wurm drin.

Nagios selbst besitzt keine Administrationsoberfläche. Gute Dienste leistet aber das eigenständige PHP-Programm „Nconf“. Hier lassen sich sehr detaillierte Einstellungen vornehmen und Abhängigkeiten definieren.

Der Alarmfall

Antwortet ein Check nicht mit der erwarteten Meldung oder werden Grenzwerte über- oder unterschritten, startet die definierte Alarmierungslogik. Bei unkritischen Ereignissen kann das eine Benachrichtigung per E-Mail zu den üblichen Bürozeiten sein. Beispiel: Der Export für das Google Merchant Center (früher Google Base) wurde nicht vollständig übertragen. Dies ist ärgerlich, erfordert jedoch kein sofortiges Eingreifen, da keine unmittelbaren Beeinträchtigungen entstehen.

In dringenden Fällen kann Nagios über ein SMS-Gateway einen oder mehrere definierte Kontakte alarmieren – wenn also die Datenbank ausfällt und damit die Website nicht mehr erreichbar ist und sofortiges Handeln notwendig ist.

Definierbare Eskalationsstufen sorgen dafür, dass kein Problem unbemerkt bleibt. Reagiert der zuständige Mitarbeiter nicht in der vereinbarten Zeit und markiert den Alarm zumindest als „Zur Kenntnis genommen“, können weitere Kontakte benachrichtigt werden. Auch in diesem Punkt lässt sich Nagios flexibel an die vorhandenen Strukturen anpassen.

Des Weiteren kann Nagios sogenannte Event Handler ausführen. Das können Skripte oder Befehle sein, die z. B. ein Datenbank-Reparatur-Tool aktivieren oder den Webserver neu starten.

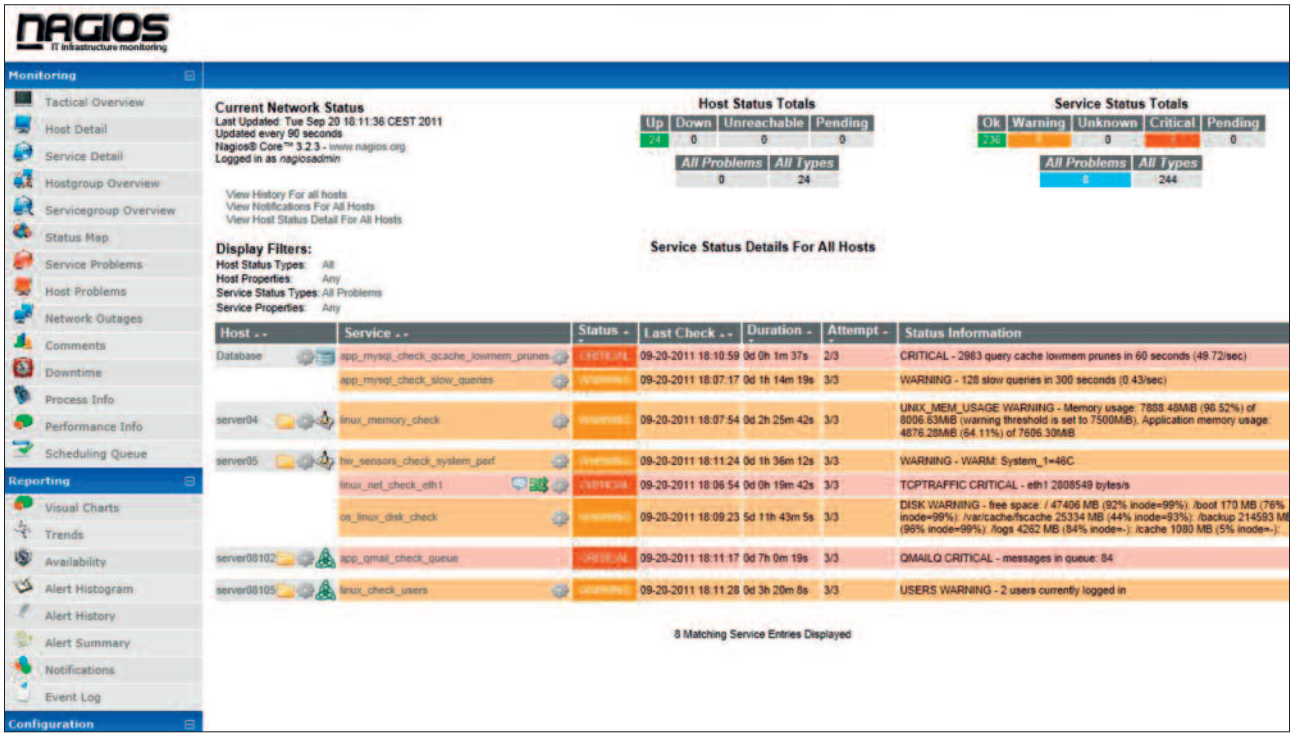

Abbildung 2: Ein gutes Tool liefert verlässliche Fehlermeldungen und -beschreibungen

Was wird überwacht?

Was genau überwacht werden muss, hängt natürlich von der individuell vorhandenen IT-Infrastruktur ab. Vorrangiges Ziel ist selbstverständlich, die Erreichbarkeit der Website sicherzustellen. Der wichtigste Check prüft darum logischerweise die Anzeige, die auch ein Besucher in seinem Browser erhalten würde. Doch es gibt noch zahlreiche weitere Parameter, die für einen ungestörten Betrieb ausschlaggebend und damit für die Überwachung relevant sind.

Betrachten wir einen typischen Webserver, so finden wir zum Beispiel:

» **CPU-Auslastung:** Wie hoch ist die Anforderung an die Rechenleistung? Reicht die Kapazität aus, um alle Aufgaben erledigen zu können?

» **Hauptspeicher-Auslastung:** Wenn zu wenig Speicher zur Verfügung steht und der Server anfängt, Daten auf die Festplatten auszulagern, sinkt die Geschwindigkeit rapide. Umgekehrt kann ungenutzter Hauptspeicher einer Anwendung zugeordnet und damit u. U. ein Performance-Zuwachs erzielt werden.

» **Festplatten-Auslastung:** Logfiles, User-Uploads, schlecht programmierte Proxy-Server und andere Prozesse können unbemerkt die Festplatte füllen und damit einen Ausfall provozieren.

» **Festplatten-Status:** Moderne Festplatten verfügen über Selbstdiagnose-Werkzeuge. Steigt die Anzahl der unbrauchbaren Sektoren, ist ein Ausfall der kompletten Festplatte zu erwarten.

» **CPU- und Gehäuse-Temperatur:** Welche Temperaturen herrschen im Rechner, laufen noch alle Lüfter? Zu viel Wärme verringert die Zuverlässigkeit und die Lebensdauer der verbauten Hardware.

Auf der Ebene des Betriebssystems sind wichtige Informationen:

» **Verfügbare Updates:** Existiert eine neue Sicherheitslücke, die mit einem Patch gestopft werden muss, oder ganz generell neue Versionen der installierten Software-Pakete?

» **Eingeloggte User:** Sind mehr Administratoren auf dem Server angemeldet als üblich? Oder laufen ungewöhnliche oder besonders viele Prozesse? Ist der Server vielleicht kompromittiert?

» **Automatischer Prozesse:** Werden alle Hintergrundprozesse vollständig ausgeführt?

Und auf Ebene der ausgeführten Programme interessiert uns vor allem der Betriebszustand, also ob alle Programme innerhalb normaler Parameter arbeiten, etwa:

» **Apache:** Wie viele Prozesse sind gestartet und was tun diese?

» **Datenbank:** Sind die Performance-Werte in Ordnung, ist die Replikation aktiv, sind einzelne Abfragen zu langsam?

» **Virenscanner:** Sind die Viren-Signaturen aktuell?

» **Mailserver:** Wird die E-Mail-Warteschlange abgearbeitet oder entsteht ein Stau, etwa weil der Server auf einer Blacklist gelandet ist?

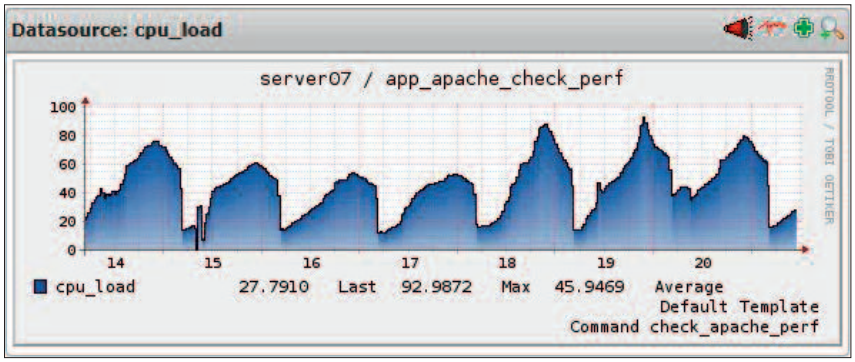

Abbildung 3: Lastspitzen, Trends und Flaschenhälse erkennen

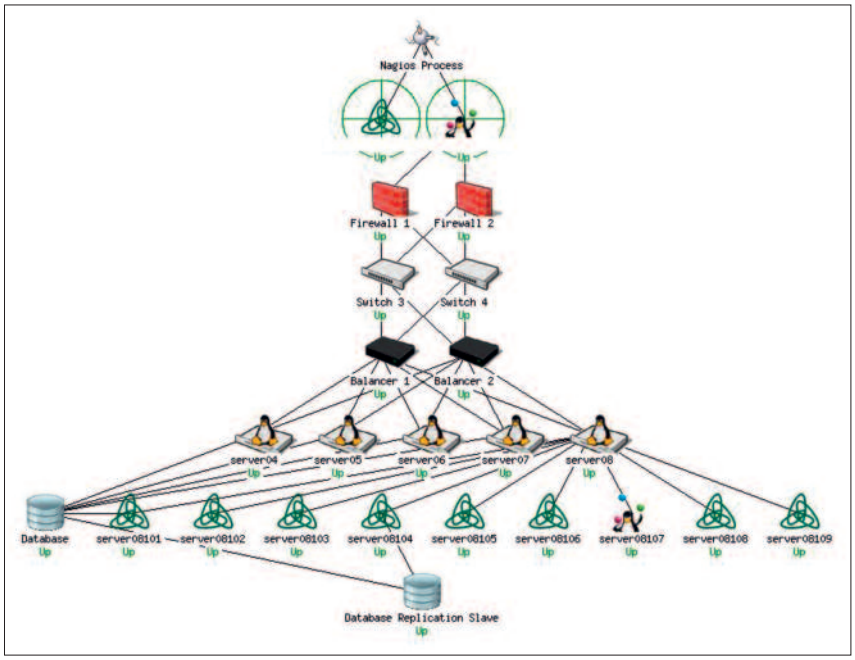

Abbildung 3: Optimal grafisch aufbereitet: Die überwachte Infrastruktur eines mittelgroßen Online-Shops

Weiterhin lassen sich natürlich auch die Zustände und kritischen Schnittstellen innerhalb einer Internet-Anwendung prüfen, zum Beispiel in einem Shop:

» Werden neue Bestellungen generiert?
» Aktualisiert der Kreditkarten-Dienstleister ausstehende Zahlungen?
» Werden Daten vom und zum Warenwirtschaftssystem übertragen?
» Erfolgt die zeitnahe Bearbeitung durch die zuständigen Mitarbeiter?

Wie wird ein Monitoring aufgebaut?

Zunächst ist eine Analyse der vorhandenen IT-Infrastruktur notwendig. Dazu gehören eine Inventarisierung der eingesetzten Hard- und Software, eine Dokumentation der vorhandenen Abhängigkeiten und die Definition der bei einem Check zu erwarteten Antworten bzw. Messbereiche.

Anschließend erfolgt die Installation und Konfiguration der Nagios-Instanz und der benötigten Plug-ins.

Falls notwendig, erfolgt die Programmierung individueller Checks. Dies kann in jeder beliebigen Sprache geschehen. Beliebt sind dabei Skriptsprachen wie Bash, Python, Perl, aber auch echte Programmiersprachen wie C.

Im laufenden Betrieb ergeben sich weitere kontinuierliche Aufgaben. Selbstverständlich müssen die angezeigten Probleme oder Fehlfunktionen behoben werden. Die aufgezeichneten Leistungswerte sind zu betrachten, zu interpretieren und zu bewerten, denn daraus ergeben sich unter Umständen notwendige Änderungen an der IT-Infrastruktur.

Aufgetretene Probleme sollten analysiert werden: Hat Nagios rechtzeitig reagiert, müssen die Schwellenwerte justiert werden, hat die Eskalationskette gegriffen, sind zusätzliche Checks notwendig?

Alles läuft – zusätzlicher Mehrwert?

Nagios prüft nicht nur den aktuellen Zustand, sondern speichert darüber hinaus alle abgefragten Werte in einer Datenbank. In der Administrationsoberfläche lassen sich mithilfe weiterer Plug-ins diese Werte anzeigen, grafisch aufbereiten und in einem zeitlichen Verlauf darstellen. Mehrere Werte können bequem miteinander verglichen werden. Verschiedene Reporting-Funktionen erzeugen Übersichten und Zusammenfassungen.

Welche Schlüsse lassen sich daraus ziehen? In jeder Web-Traffic-Statistik sind Last-Spitzen erkennbar – also die Zeitpunkte, zu denen sich die meisten Besucher auf einer Website bewegen. Aber was bedeutet das? Ist der Server zu diesem Zeitpunkt wirklich ausgelastet? Stößt die Netzwerkverbindung an ihre Kapazitätsgrenzen? Reagiert der Shop vielleicht langsamer? Oder ist für die Kunden tatsächlich keinerlei Beeinträchtigung zu bemerken? Mithilfe der Nagios-Auswertung lassen sich diese Fragen beantworten.

Trends und Flaschenhälse

Durch die Analyse der aufgezeichneten Leistungsdaten über einen längeren Zeitraum hinweg lassen sich Trends erkennen.

Offensichtliche Trends sind etwa die Entwicklung von Traffic und CPU-Aus-

lastung. Wie groß waren diese Werte letztes Jahr, wie war die Entwicklung in der Zwischenzeit, welche Schlüsse lassen sich daraus im Hinblick auf die zukünftigen Anforderungen ziehen? Das Vorweihnachtsgeschäft ist meist die Periode der intensivsten Nutzung eines Online-Shops. Wenn der Server im Vorjahr an seine Grenzen gestoßen ist, im Jahresverlauf der durchschnittliche Traffic um, sagen wir, 20 % gewachsen ist, wäre genau jetzt der Zeitpunkt, über ein Upgrade nachzudenken.

Ein Beispiel für einen weniger offensichtlichen Trend ist die durchschnittliche Antwortzeit der Datenbank. Ist diese in den vergangenen Monaten kontinuierlich gestiegen, ohne dass dies durch erhöhten Traffic begründet ist? Liegt hier möglicherweise ein Problem vor?

Flaschenhälse sind Engstellen, die das übrige System unnötig ausbremsen. Auch diese lassen sich mit den ermittelten Nagios-Daten identifizieren. Dies können z. B. eine langsame Festplatte, eine falsch konfigurierte Netzwerkkarte, eine zu geringe Speicherzuteilung oder eine ungünstig gewählte Software-Einstellung sein, was im schlimmsten Fall eine spürbare Leistungsbremse bedeutet.

Oft genug lassen sich diese Probleme mit einfachen Mitteln beheben. Eventuell benötigt nur die Datenbank etwas mehr zugeteilten Cache, um auch bei Lastspitzen performant arbeiten zu können. Oder eine SQL-Abfrage ist falsch programmiert und verwendet keinen Index – was bei der Entwicklung gar nicht aufgefallen ist, aber im Live-Betrieb ein Problem darstellt.

Es muss also vielleicht gar nicht ein neuer, größerer Server angeschafft werden.

Fazit

Richtig eingesetzt haben Sie mit Server-Monitoring ihre Web-Infrastruktur vollständig unter Kontrolle. Es entsteht ein spürbarer Zugewinn an Ausfallsicherheit. Probleme werden, soweit möglich, antizipiert und können rechtzeitig, d. h. bevor eine ungeplante Downtime eintritt, behoben werden. Natürlich kann ein Ausfall einer Komponente nie vollständig verhindert werden – aber dies wird zeitnah bemerkt und kann umgehend angegangen werden, im günstigsten Fall, bevor dadurch eine Beeinträchtigung für den Kunden entsteht.¶