



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 600 04 886 T2 2004.07.08**

(12)

Übersetzung der europäischen Patentschrift

(97) **EP 1 242 873 B1**

(51) Int Cl.⁷: **G06F 9/00**

(21) Deutsches Aktenzeichen: **600 04 886.1**

(86) PCT-Aktenzeichen: **PCT/US00/34248**

(96) Europäisches Aktenzeichen: **00 986 503.1**

(87) PCT-Veröffentlichungs-Nr.: **WO 01/046799**

(86) PCT-Anmeldetag: **18.12.2000**

(87) Veröffentlichungstag
der PCT-Anmeldung: **28.06.2001**

(97) Erstveröffentlichung durch das EPA: **25.09.2002**

(97) Veröffentlichungstag
der Patenterteilung beim EPA: **27.08.2003**

(47) Veröffentlichungstag im Patentblatt: **08.07.2004**

(30) Unionspriorität:
468493 20.12.1999 US

(84) Benannte Vertragsstaaten:
DE, GB

(73) Patentinhaber:
Unisys Corp., Blue Bell, Pa., US

(72) Erfinder:
**SIEVERT, A., James, Lino Lakes, US; VALLEVAND,
K., Mark, Lino Lakes, US**

(74) Vertreter:
Schieber und Kollegen, 80469 München

(54) Bezeichnung: **SYSTEM UND VERFAHREN ZUM BEREITSTELLEN EINER SAMMLUNG VON WIEDERVERWENDBAREN FÄDEN ZUM BEHANDELN VON GEPUFFERTEN AUFGABEN**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung**ERFINDUNGSGEBIET**

[0001] Die vorliegende Erfindung betrifft allgemein Fäden für eine Computerprogrammausführung und insbesondere die Verwaltung eines Pools von Fäden zur Ausführung von in Schlange eingesetzten Arbeitsgegenständen.

HINTERGRUND DER ERFINDUNG

[0002] Ein Faden wird allgemein als leicht gewichtiger Fluss der Steuerung innerhalb eines Prozesses angesehen, worin ein Prozess einen schwergewichtigen Fluss der Steuerung darstellt. Ein Prozess ist dahingehend schwergewichtig, dass das Betriebssystem einen dezidierten Speicherbereich für den Programmcode und die Daten zuordnet, die vom Prozess verwendet werden, wohingegen ein Faden darin leicht gewichtig ist, dass ein Faden innerhalb des Speicherbereichs eines Prozesses existiert. Einige Prozesse haben mehrere Fäden, so dass mehrere Fäden gleichzeitig innerhalb eines Prozesses ablaufen können.

[0003] Fäden werden im allgemeinen mit der Unterstützung von einem Betriebssystem erzeugt und für gewöhnlich ausgeführt, indem der Faden mit einem bestimmten Betrieb gekoppelt wird. Das bedeutet, dass ein Faden zum Zweck einer Durchführung eines bestimmten Betriebs erzeugt wird. Nachdem der Betrieb abgeschlossen ist, kann der Faden zerstört oder beim Durchführen desselben Betriebs mit vielleicht anderen Eingabedaten wiederverwendet werden.

[0004] Das Erzeugen und Initiieren von neuen Fäden kann hinsichtlich des Verbrauchs von Betriebssystemressourcen teuer sein. Jedesmal, wenn ein neuer Faden benötigt wird, muss die Steuerung an das Betriebssystem weitergegeben werden, um den Faden zu erzeugen und irgendwelche für das Verwalten des Fadens erforderlichen Strukturen zu errichten und zu erzeugen. Solchermaßen erzeugen einige Applikationen mehrere Fäden zum Durchführen desselben Betriebs innerhalb eines Prozesses. Jedoch sind die Fäden für die spezifischen Betriebe dezidiert, für die sie erzeugt wurden, was hinsichtlich der Ausnutzung von Ressourcen ineffizient sein kann. Beispielsweise kann eine erste Gruppe von Fäden erzeugt und für eine erste Funktion eines Systems dezidiert sein, und eine zweite Gruppe von Fäden kann erzeugt und für eine zweite Funktion des Systems dezidiert sein. Wenn die Nachfrage für die erste Funktion niedrig ist, d. h. worin nur ein wenige Fäden der ersten Gruppe von Fäden aufgerufen werden, und die Nachfrage für die zweite Funktion hoch ist, wobei es mehr Nachfrage für die zweite Funktion gibt, als es verfügbare Fäden gibt, dann kann die Leistung des Systems durch die begrenzte für die Durchführung der zweiten Funktion verfügbare Anzahl von Fäden eingeschränkt werden. Mit anderen

Worten können die Nachfragen für eine von der zweiten Funktion durchgeführten Arbeit aufgrund der begrenzten mit der zweiten Funktion verknüpften Anzahl von Fäden verzögert werden.

[0005] Das vorherige Szenarium veranschaulicht eine Situation, worin die Leistung selbst dann verschlechtert werden kann, wenn es eine für das Erfüllen der Nachfrage verfügbare Verarbeitungsfähigkeit geben kann. Spezifisch sind die mit der ersten Funktion verknüpften Fäden im Wartezustand, während die mit der zweiten Funktion verknüpften Fäden vollständig benutzt werden.

[0006] Ein System zum Verwalten von Fäden wird im U.S.-Patent 5.504.898 von Klein offenbart. Das offenbarte System stellt eine Umgebung mit Fäden bereit, indem Faden-bezogene Stapeljobs verwendet werden, um Fadenaufgaben anzunehmen und sie einzeln den spezialisierten Stapeljobs anzutragen, die die Aufgaben durchführen. Rücksprungdaten werden von den Faden-bezogenen Stapeljobs an einer gemeinsamen Schlange angelegt, und es wird veranlasst, dass die verwendete Applikation an der gemeinsamen Schlange wartet. Obwohl das 5.504.898-Patent ein System offenbart, das in der Lage ist, einen Pool der aktiven Faden-bezogenen Stapeljobs aufrechtzuerhalten, werden diese Jobs nicht auf eine Art und Weise verwaltet, die allen oben erörterten Leistungsaufgaben begegnet.

[0007] Ein Verfahren und ein System, das den zuvor erwähnten Problemen und anderen betreffenden Problemen begegnet, ist daher wünschenswert.

ZUSAMMENFASSUNG DER ERFINDUNG

[0008] Ein System und ein Verfahren zum Verwalten eines Pools von Fäden wird zum Ausführen von Fadenbetrieben bereitgestellt. Eine Arbeitsschlange wird bereitgestellt, um die gesteuerte Zuweisung von Faden-bezogenen Ressourcen zur Ausführung einer Mehrzahl an Fadenbetrieben zu unterstützen. Ein Faden-Pool wird instantiiert, der eine Mehrzahl von Fäden einschließt, und eine Arbeitsschlange, die Programm-steuerbare Zustände hat, wird instantiiert. Fadenbetriebe werden in die Arbeitsschlange einge- reiht, wenn sich die Arbeitsschlange in einem laufen- den oder unterbrochenen Zustand befindet, und wenn sich die Arbeitsschlange in einem gestoppten Zustand befindet, werden Anforderungen zum Einrei- hen in die Schlange von Fadenbetrieben verweigert. In anderen Ausführungsformen können vorüberge- hende Fäden in den Faden-Pool eingeführt werden, um einem zeitweiligen Bedarf an zusätzlichen Fa- den-Ressourcen zu begegnen.

[0009] Es ist nicht beabsichtigt, dass die obige Zu- sammenfassung der vorliegenden Erfindung jede of- fenbarte Ausführungsform der vorliegenden Erfin- dung beschreibt. Die Figuren und die detaillierte Be- schreibung, die folgen, stellen zusätzliche beispiel- hafte Ausführungsformen und Aspekte der vorliegen- den Erfindung bereit.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

DETAILLIERTE BESCHREIBUNG

[0010] Weitere Aspekte und Vorteile der Erfindung werden bei der Durchsicht der detaillierten Beschreibung und unter Bezugnahme auf die Zeichnungen ersichtlich, in denen:

[0011] **Fig. 1** ein Funktionsblockdiagramm eines Systems zum Bereitstellen eines Pools von wiederverwendbaren Fäden ist, damit in Übereinstimmung mit einer Ausführungsform der Erfindung Arbeitsgegenstände durchgeführt werden;

[0012] **Fig. 2** ein Zustandsdiagramm ist, das mögliche Zustände des Faden-Pools und Übergänge zwischen den Zuständen zeigt;

[0013] **Fig. 3** das Verhältnis zwischen den **Fig. 3A** und **3B** zeigt, die zusammen ein Klassendiagramm einer Ausführungsform einer Klassenanordnung zum Bereitstellen wiederverwendbarer Fäden und einer Arbeitsschlange der Fadenbetriebe umfassen;

[0014] **Fig. 4** ein Objekt-Interaktionsdiagramm ist, das die Einreihung eines Fadenbetriebs in die Arbeitsschlange darstellt, wenn sich der Faden-Pool in einem laufenden oder unterbrochenen Zustand befindet;

[0015] **Fig. 5** ein Objekt-Interaktionsdiagramm ist, das den Übergang des Faden-Pools von einem laufenden zu einem unterbrochenen Zustand darstellt;

[0016] **Fig. 6** ein Objekt-Interaktionsdiagramm ist, das den Übergang des Faden-Pools von einem laufenden zu einem gestoppten Zustand darstellt;

[0017] **Fig. 7** ein Objekt-Interaktionsdiagramm ist, das den Übergang des Faden-Pools von einem unterbrochenen zu einem laufenden Zustand darstellt;

[0018] **Fig. 8** ein Objekt-Interaktionsdiagramm ist, das den Übergang des Faden-Pools von einem unterbrochenen zu einem gestoppten Zustand darstellt;

[0019] **Fig. 9** ein Objekt-Interaktionsdiagramm ist, das den Übergang des Faden-Pools von einem gestoppten zu einem unterbrochenen Zustand darstellt;

[0020] **Fig. 10** ein Objekt-Interaktionsdiagramm ist, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem unterbrochenen Zustand befindet;

[0021] **Fig. 11** ein Objekt-Interaktionsdiagramm ist, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem laufenden Zustand befindet;

[0022] **Fig. 12** ein Objekt-Interaktionsdiagramm ist, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem gestoppten Zustand befindet und die Arbeitsschlange nicht leer ist; und

[0023] **Fig. 13** ein Objekt-Interaktionsdiagramm ist, das die Ausführung eines Fadens zeigt, wenn sich der Faden-Pool in einem gestoppten Zustand befindet und die Arbeitsschlange leer ist.

[0024] Obwohl die Erfindung für verschiedene Modifikationen und alternative Formen zugänglich ist, wurden ihre besonderen Ausführungsformen exemplarisch in den Zeichnungen gezeigt und werden detailliert hierin ebenfalls exemplarisch beschrieben werden.

[0025] Es wird angenommen, dass die vorliegende Erfindung an eine Vielfalt an Systemen anlegbar ist, um Fäden und die von den Fäden durchgeführte Arbeit zu verwalten. Es wurde befunden, dass die vorliegende Erfindung besonders anwendbar und vorteilhaft ist, wenn sie in einer Objekt-orientierten Programmiersprache implementiert wird. Obwohl die vorliegende Erfindung nicht so eingeschränkt ist, wird mithilfe der Klassendefinitionen, die verwendet werden, um die Erfindung zu implementieren, eine Einschätzung der vorliegenden Erfindung gezeigt.

[0026] **Fig. 1** ist ein Funktionsblockdiagramm eines Systems zur Bereitstellung eines Pools von wiederverwendbaren Fäden für die Durchführung von Arbeitsgegenständen in Übereinstimmung mit einer Ausführungsform der Erfindung. Das System **10** schließt eine Arbeitsschlange **12**, einen Faden-Pool **14**, einen Arbeitsschlangen-Controller **16** und einen Faden-Pool-Manager **18** ein. Fadenbetriebe **20** führen Aufrufe für den Arbeitsschlangen-Controller **16** aus, um in der Arbeitsschlange **12** eingereiht zu werden.

[0027] Die Arbeitsschlange **12** unterstützt ein Verfahren zum Verwalten einzelner Arbeitsgegenstände, die von irgendeinem Faden **22** im Faden-Pool **14** ausgeführt werden können. In einer Ausführungsform sind die in die Schlange eingereihten Arbeitsgegenstände WIN32 API-Aufrufe, die einzeln von Fadenbetrieben **24** umwickelt und dem Windows NT-Betriebssystem-("NT")-E/A-Durchführungsport zur Verfügung gestellt werden.

[0028] Die Arbeitsschlange **12** kann sich in einem von drei Zuständen befinden: gestoppt, unterbrochen und laufend. Wenn die Arbeitsschlange instantiiert wird, befindet sie sich anfänglich im gestoppten Zustand. Die für die Einreihung in die Arbeitsschlange **12** gesuchten Fadenbetriebe werden abgelehnt, wenn sich die Arbeitsschlange im gestoppten Zustand befindet. Wenn sich die Arbeitsschlange **12** im unterbrochenen Zustand befindet, sind die Fäden **22** im Wartezustand und führen keine Fadenbetriebe **24** durch. Jedoch können neue Fadenbetriebe in der Arbeitsschlange **12** eingereiht werden, wenn sich die Arbeitsschlange im unterbrochenen Zustand befindet. Wenn die Arbeitsschlange **12** läuft, führen Fäden **22** Fadenbetriebe **24** aus, und es können neue Fadenbetriebe die die Schlange eingereiht werden.

[0029] Der Faden-Pool-Manager **18** instantiiert Fäden **22** im Faden-Pool **14**, indem Start-Faden-Anforderungen verwendet werden, die an den Arbeitsschlangen-Controller **16** ausgegeben werden. Fäden **22** sind für die Ausführung der Fadenbetriebe **24** verfügbar. Da die Erzeugung von Fäden hinsichtlich der Betriebssystem-Ressourcen-Erfordernisse teuer ist, stellt ein Pool von Fäden die Fähigkeit bereit, neue Fadenbetriebe auszuführen, ohne auf die Erzeugung eines neuen Fadens warten zu müssen.

[0030] Der Arbeitsschlangen-Controller **16** sorgt für

die Fähigkeit, den Arbeitsfluss (d. h. die Ausführung der Fadenbetriebe **24**) eng mit dem Faden-Pool-Manager **18** zu steuern. Der Faden-Pool **14** kann über den Arbeitsschlangen-Controller **16** vom Faden-Pool-Manager **18** z. B. gestartet, gestoppt und unterbrochen werden. Wenn der Faden-Pool **14** gestoppt wird, wird der Arbeitsschlangen-Controller **16** die Fadenbetriebe nicht mehr länger in die Arbeitsschlange **12** einreihen, noch nicht begonnene Fadenbetriebe **24** werden aus der Schlange genommen und nicht ausgeführt, und in Ausführung befindliche Fadenbetriebe dürfen zum Abschluss gebracht werden. Wenn der Faden-Pool **14** unterbrochen wird; wird der Arbeitsschlangen-Controller damit fortfahren, die Fadenbetriebe **20** in der Arbeitsschlange **12** einzureihen, und Fadenbetriebe **24**, die noch nicht gestartet sind, werden nicht aus der Schlange herausgenommen, bis der Faden-Pool-Zustand auf laufend geändert wird.

[0031] In einer anderen Ausführungsform der Erfindung wird eine Mehrzahl von Instanzen des Systems **10** erzeugt, um mehrere Arbeitsschlangen und Faden-Pools bereitzustellen. Zum Beispiel können verschiedene Applikationen unterschiedliche Ressourcen-Erfordernisse haben. Solchermaßen kann der mit einer Applikation verknüpfte Faden-Pool kleiner sein als der mit einer anderen Applikation verknüpfte. [0032] Der Arbeitsschlangen-Controller **16** stellt auch die Fähigkeit bereit, um vorübergehend Fäden zum Faden-Pool **14** hinzuzufügen. Dieses Merkmal kann im Zusammenhang mit unabhängigen Arbeitsschlangen und Faden-Pools für verschiedene Applikationen nützlich sein. Wenn z.B. während einer bestimmten Zeitspanne eine erste Applikation einen geringeren Bedarf an der Berechnung von Ressourcen als eine zweite Applikation hat, dann können einigen der Fäden in dem mit der ersten Applikation verknüpften Faden-Pool zeitweilig am Faden-Pool der zweiten Applikation angehängt werden. Wenn die Zeitspanne verstrichen ist, können die Fäden vom Faden-Pool der zweiten Applikation entfernt und zum Faden-Pool der ersten Applikation zurückgegeben werden.

[0033] **Fig. 2** ist ein Zustandsdiagramm, das mögliche Zustände des Faden-Pools und Übergänge zwischen den Zuständen zeigt. Die Zustände des Faden-Pools **14** sind gestoppt, unterbrochen und laufend. Die Verfahren innerhalb eines jeden der Blöcke sind die Namen der Verfahren, die bewirken, dass die Arbeitsschlange in den verknüpften Zustand übergeht. Die Konvention für diese Diagramme ist, den anfänglichen Zustand als vom Startknoten aus eintretenden zu zeigen.

[0034] Die Stopp-Schlange- und Wiederaufnahme-Schlange-Verfahren bewirken, dass die Arbeitsschlange in den gestoppten Zustand übergeht, die Unterbreche-Schlange- und das Start-Schlange-Verfahren bewirken, dass die Arbeitsschlange in den unterbrochenen Zustand übergeht, und das Wiederaufnahme-Schlange-Verfahren bewirkt, dass die Ar-

beitsschlange in den laufenden Zustand übergeht. Wenn sich der Faden-Pool **14** im gestoppten Zustand befindet, bewirkt das Aufrufen des Start-Schlange-Verfahrens, dass der Faden-Pool vom gestoppten Zustand zum unterbrochenen Zustand übergeht. Das Aufrufen von entweder dem Wiederaufnahme-Schlange-Verfahren oder dem Stopp-Schlange-Verfahren, wenn sich der Faden-Pool **14** im unterbrochenen Zustand befindet, bewirkt; dass der Faden-Pool im gestoppten Zustand bleibt.

[0035] Wenn sich der Faden-Pool **14** im unterbrochenen Zustand befindet, bewirkt das Aufrufen des Stopp-Schlange-Verfahrens den Faden-Pool zum Übergang in den gestoppten Zustand. Das Aufrufen des Wiederaufnahme-Schlange-Verfahrens veranlasst den Faden-Pool zum Übergang vom unterbrochenen Zustand zum laufenden Zustand.

[0036] Der Faden-Pool **14** bleibt im unterbrochenen Zustand, wenn das Unterbreche-Schlange-Verfahren aufgerufen wird, während sich der Faden-Pool gegenwärtig im unterbrochenen Zustand befindet.

[0037] Wenn sich der Faden-Pool **14** im laufenden Zustand befindet, veranlasst das Aufrufen des Stopp-Schlange-Verfahrens den Faden-Pool zum Übergang in den gestoppten Zustand, und das Aufrufen des Unterbreche-Schlange-Verfahrens veranlasst den Faden-Pool zum Übergang in den unterbrochenen Zustand. Das Aufrufen des Wiederaufnahme-Schlange-Verfahrens, wenn sich der Faden-Pool in den laufenden Zuständen befindet, veranlasst den Faden-Pool zum Bleiben im laufenden Zustand.

[0038] Die Zustand-Übergangslinien schließen durch Konvention Markierungen ein, die auf die Funktion hinweisen, die den Zustand-Übergang bewirkt, der von Aktionen gefolgt wird, die durchgeführt werden, wenn der Übergang erfolgt. Der Übergang vom unterbrochenen Zustand zum gestoppten Zustand wird z. B. durch die Stopp-Schlange-Funktion veranlasst. Die Änderung im Zustand führt dazu, dass die in Schlangen eingereihten Arbeitsfäden für die Beendigung aus der Schlange freigegeben werden. Weitere Details werden in den Figuren, die folgen, bereitgestellt.

[0039] **Fig. 3** zeigt das Verhältnis zwischen den **Fig. 3A** und **3B**, die zusammen ein Klassendiagramm einer Ausführungsform einer Klassenanordnung zum Verwalten eines Pools von wiederverwendbaren Fäden und eine Arbeitsschlange der Fadenbetriebe umfassen. Die Figur verwendet eine herkömmliche Objekt-Modellier-Technik(OMT)-Bezeichnung, um die Klassen und Klassenverhältnisse zu bezeichnen.

[0040] Ein Objekt des Klassen-Arbeitsschlangen-Faden-Pools verwaltet Objekte der Fadenklasse, und Objekte des Klassen-Arbeitsschlangen-Zustand-Controllers erben vom Fadenbetrieb der abstrakten Klasse. Die Fäden und Fadenbetriebsklassen werden in der ebenfalls anhängigen Patentanmeldung detailliert beschrieben.

[0041] Ein Arbeitsschlangen-Objekt verfügt über ein

Arbeitsschlangen-Zustand-Controller-Objekt, das ein Arbeitsschlangen-Faden-Pool-Objekt und ein Arbeitsschlangen-Implementierung-Objekt hat. Die Pfeilspitze auf der Linie, die die Arbeitsschlangen-Klasse mit dem Arbeitsschlangen-Zustand-Controller verkettet, weist darauf hin, dass es sich um eine Ein-Wege-Verkettung handelt. Das bedeutet, dass die Arbeitsschlange auf den Arbeitsschlangen-Zustand-Controller verweisen kann, aber der Controller nicht auf die Arbeitsschlange verweisen kann. Wenn eine Verkettung keine Pfeilspitzen hat (z. B. zwischen dem Arbeitsschlangen-Faden-Pool und dem Arbeitsschlangen-Zustand-Controller), ist sie bidirektional, wobei Objekte auf jeder Seite der Verkettung aufeinander verweisen können.

[0042] Ein Arbeitsschlangen-Zustand-Controller-Objekt hat Objekte des Arbeitsschlangen-Gestoppt-Zustands, Arbeitsschlangen-Unterbrochen-Zustands und Arbeitsschlangen-Laufend-Zustands. Jedes der verschiedenen Zustand-Objekte erbt vom der Klasse Arbeitsschlangen-Zustand, der von einem Objekt der Arbeitsschlangen-Zustand-Controller-Klasse aufrechterhalten wird.

[0043] Die Arbeitsschlangen-Zustände-Klasse kapselt die aufgezählten Zustand-Werte von gestoppt, laufend, unterbrochen und unbekannt ein.

[0044] Die Attribute und Verfahren der unterschiedlichen Klassen werden in den Paragraphen, die folgen, beschrieben.

Arbeitsschlangen-Klasse

[0045] Die Arbeitsschlangen-Klasse sorgt für die Schnittstelle, mittels der die Fadenbetriebe in der Arbeitsschlange **12** eingereiht werden und der Faden-Pool **14** verwaltet wird.

[0046] Das Anhängen-Faden-Verfahren der Arbeitsschlangen-Klasse kann verwendet werden, um einen Faden zum Faden-Pool **14** hinzuzufügen. Solchermaßen können zu Zeiten, in denen mehr Arbeit durchgeführt werden muss (mittels Fadenbetrieben in der Arbeitsschlange **12**), zusätzliche Fäden zum Faden-Pool **14** hinzugefügt werden, indem das Anhängen-Faden-Verfahren verwendet wird. Der Faden bleibt im Faden-Pool **14**, bis das Stopp-Schlange-Verfahren durchgeführt wird. Das Stopp-Schlange-Verfahren entleert die Arbeitsschlange **12** von Fadenbetrieben **24**, gibt diejenigen Fäden **22** zurück, die mittels der Verwendung des Anhängen-Faden-Verfahrens zum Faden-Pool **14** hinzugefügt wurden, um Punkte außerhalb des Systems **10** zu steuern, und unterbricht diejenigen Fäden **22**, die mittels Verwendung des Erzeuge-Fäden-Verfahrens der Arbeitsschlange-Faden-Pool-Klasse erzeugt wurden.

[0047] Das Start-Schlangen-Verfahren wird verwendet, um den Zustand der Arbeitsschlange **12** vom gestoppten Zustand auf den unterbrochenen Zustand zu ändern.

[0048] Das Stopp-Schlange-Verfahren wird verwendet, um den Zustand der Arbeitsschlange **12** auf den

gestoppten Zustand zu ändern.

[0049] Das Unterbreche-Schlange-Verfahren wird verwendet, um den Zustand der Arbeitsschlange **12** vom laufenden Zustand auf den unterbrochenen Zustand zu ändern.

[0050] Das Wiederaufnahme-Schlange-Verfahren wird verwendet, um den Zustand der Arbeitsschlange **12** vom unterbrochenen Zustand auf den laufenden Zustand zu ändern.

[0051] Das Schlangenarbeitsverfahren wird verwendet, um einen Fadenbetrieb **20** zur Arbeitsschlange **12** hinzuzufügen.

[0052] Das Hole-Arbeitsschlangen-Zustand-Verfahren gibt den aktuellen Zustand zur Arbeitsschlange **12** zurück.

[0053] Das Hole-Schlangen-Tiefen-Verfahren gibt die Anzahl der Fadenbetriebe in der Arbeitsschlange **12** zurück, die auf ihre Ausführung warten.

[0054] Das Wartezustand-Synchronisier-Verfahren fordert das entsprechende Verfahren der Arbeitsschlangen-Zustand-Controller-Klasse zu warten auf, bis die Schlange im aktuellen Zustand im Wartezustand ist. Die Arbeitsschlange ist im Wartezustand; wenn es in der Arbeitsschlange keine eingereihten Fadenbetriebe gibt.

Arbeitsschlangen-Faden-Pool-Klasse

[0055] Die Arbeitsschlangen-Faden-Pool-Klasse schließt Attribute und Verfahren zum Verwalten eines Faden-Pools **14** ein.

[0056] Max-Fäden sind die im Faden-Pool **14** erlaubte maximale Anzahl von Fäden.

[0057] Die Anzahl der Fäden ist die im Faden-Pool **14** vorhandene Anzahl von Fäden.

[0058] Das Faden-Pool-Attribut verweist auf die Objekte der Klasse Faden.

[0059] Das Fadenbetrieb-Attribut verweist auf den Betrieb, den ein Faden ausführt.

[0060] Das Faden-Speicher-Manager-Attribut verweist auf den Speicherbereich, in dem die Faden-Objekte zugeordnet werden.

[0061] Das Zuweise-Verfahren implementiert die Zuweisung von einem Objekt zu einem anderen. Es ist auch als Zuweisungsoperator bekannt. In diesem Fall wird es erklärt, aber nicht bestimmt (keine Code-Implementierung), da die Zuweisung nicht unterstützt wird. Wenn ein Programmierer versuchen würde, ein Objekt einem anderen zuzuordnen, würde der Compiler einen Fehler melden, anstatt den Fehler-Zuweisungsoperator zu verwenden.

[0062] Das Start-Fäden-Verfahren startet ausdrücklich die Fäden im Pool.

[0063] Das Warten-Auf-Vollendung-Verfahren wird aufgerufen, um den Aufrufer zu unterbrechen, bis alle Fäden im Pool abgeschlossen sind.

[0064] Das Erzeuge-Fäden-Verfahren ruft das Betriebssystem dazu auf, die Fäden des Faden-Pools **14** zu erzeugen.

[0065] Das Zerstöre-Fäden-Verfahren ruft das Be-

triebssystem auf, die Fäden des Faden-Pools **14** freizugeben.

Arbeitsschlangen-Zustand-Controller-Klasse

[0066] Die Arbeitsschlangen-Zustand-Controller-Klasse schließt Attribute und Verfahren für die Schnittstellenverbindung zwischen der Faden-Pool-Implementierung und der Arbeitsschlangen-Implementierung ein.

[0067] Der Arbeitsfaden-Zählwert ist die Anzahl der dem Faden-Pool zugeordneten "Arbeiter"-Fäden. Ein "Arbeiter"-Faden ist ein Ausführungsfaden, worin der Begriff Arbeiter-Faden verwendet wird, um Fäden im Pool von anderen Fäden zu unterscheiden.

[0068] Der Zählwert der angehängten Fäden zählt die Anzahl der Fäden im Faden-Pool **14**, die aus dem Anhängen-Faden-Verfahren hervorgeht.

[0069] Das Kein-Betrieb-Attribut ist ein Betrieb, der keine Arbeit durchführt und der im Windows NT-Betriebssystem-E/A-Durchführungspunkt eingereiht wird. Kein-Betriebsobjekte, die am E/A-Durchführungspunkt in die Schlange eingereiht werden, bewirken, dass die Fäden aufwachen, die Tue-Nichts-Betriebe durchführen und ein weiteres in die Schlange eingereihtes Kein-Betrieb-Objekt aus der Schlange auswerfen.

[0070] Das Betrieb-Hauptverfahren ruft das aus der Arbeitsschlange **12** herausgenommene Fadenbetrieb-Objekt auf.

[0071] Das Anhängen-Faden-Verfahren erklärt einen C-Faden mit dem aktuellen Objekt initialisiert, inkrementiert den Zählwert der angehängten Fäden und hängt das Faden-Objekt an den aktuellen Faden an. Der aktuelle Faden der Ausführung wird in ein C-Faden-Objekt eingewickelt und eingehüllt, damit es mit einer Objektorientierten Syntax und Semantik verarbeitet wird. Das Faden-Objekt wird dann in den Faden-Pool gefügt, wo es bleibt, bis die Arbeitsschlange gestoppt wird. Wenn der Faden vom Faden-Objekt befreit wird, wird die Steuerung zum Anhängen-Faden-Verfahren zurückgeführt, der Zählwert der angehängten Fäden gesenkt und die Steuerung an das Arbeitsschlange-Verfahren zurückgegeben.

[0072] Die Verfahren Starte-Schlange, Stopp-Schlange, Unterbreche-Schlange, Wiederaufnahme-Schlange, Schlangen-Arbeit, Hole-Schlängenzustand und Hole-Schlängentiefe werden von den entsprechenden Verfahren der Arbeitsschlangen-Klasse aufgerufen und rufen durch abhängig vom aktuellen Zustand der Arbeitsschlange **12** zu den Verfahren der Arbeitsschlangen-Gestoppt-, -Unterbrochen- und -Laufenden-Zustand-Klassen.

[0073] Das Abwickle-Schlängenarbeitsverfahren ruft zum Schlangenarbeitsverfahren der Arbeitsschlangen-Implementierungsklasse durch.

[0074] Das Abwickle-Ausreihungs-Arbeitsverfahren ruft zum Ausreihungs-Arbeitsverfahren der Arbeitsschlangen-Implementierungsklasse durch.

[0075] Das Erhalte-Aktuelles-Arbeitsschlangen-Zu-

stand-Verfahren gibt den Zustand der Arbeitsschlange **12** auf der Grundlage des Zustands des Arbeitsschlangen-Zustände-Objekts zurück.

[0076] Das Schalte-Zustand-Verfahren ändert den Zustand der Arbeitsschlange auf synchronisierte Weise. Das Verfahren fordert ein Synchronisierungsobjekt auf zu warten, bis irgendeine andere Zustandsänderung beendet ist. Dann wird der aktuelle Zustand verlassen (durch das Aufrufen des Verlasse-Zustand-Verfahrens im Zustand), es wird auf den neuen Zustand verwiesen, und der neue Zustand wird betreten (indem das Enter-Zustand-Verfahren im Zustand aufgerufen wird). Die Fadensynchronisierung wird dann freigegeben.

[0077] Die Start-Arbeitsfäden beginnen mit der Ausführung der Fäden im Faden-Pool **14**.

[0078] Das Freigabe-Der-In-Die-Schlange-Eingereihten-Arbeitsfäden-Verfahren reiht für jeden Faden im Pool **14** abhängig vom eingegebenen Parameter entweder ein Kein-Betrieb-Objekt oder einen NULL-Arbeit-Betrieb (z. B. eine 0 zum E/A-Durchführungspunkt) in die Schlange ein. NULL-Betriebe veranlassen, dass ein Faden endet, und ein Kein-Betrieb veranlasst einen Faden, den Kein-Betrieb auszuführen und in einen neuen Zustand zu treten.

[0079] Die Betrieb-Synch-Beginn- und Betrieb-Synch-Ende-Verfahren werden verwendet, um die Fadensynchronisierung im Schalte-Zustände-Verfahren zu beginnen und zu beenden.

[0080] Das Wartezustand-Synchronisier-Verfahren fordert den aktuellen Zustand zu warten auf, bis die Arbeitsschlange im aktuellen Zustand im Wartezustand ist.

Arbeitsschlangen-Implementierungsklasse

[0081] Die Arbeitsschlangen-Implementierungsklasse implementiert die Arbeitsschlange **12**, indem die vom Windows NT-Betriebssystem bereitgestellten E/A-Durchführungspunkte verwendet werden.

[0082] Das Schlangen-Abwickleattribut ist eine Abwicklung für das E/A-Durchführungspunkt.

[0083] Das Schlängentiefeattribut ist ein verriegelter Zählwert der Anzahl von Betrieben an der Schlange. Ein verriegelter Zählwert ist ein Wort im Speicher, das auf eine nicht teilbare Weise inkrementiert, gesenkt oder mit einem neuen Wert aktualisiert wird. Ein unabhängiges Synchronisierungsobjekt ist unnötig, da die Synchronisation in dem verriegelten Zählwert eingebaut ist.

[0084] Das Schlangenarbeitsverfahren inkrementiert das Schlängentiefeattribut und reiht den Fadenbetrieb, auf den verwiesen wird, in das E/A-Durchführungspunkt ein.

[0085] Das Schlangen-Null-Arbeit-Verfahren inkrementiert die Schlängentiefe und reiht den Wert 0 in das E/A-Durchführungspunkt ein.

[0086] Das Ausreihungs-Arbeit-Verfahren erhält einen Zeiger für einen Fadenbetrieb vom E/A-Durchführungspunkt, senkt die Schlängentiefe und gibt den

Zeiger zurück.

[0087] Das Erhalte-Schlagentiefe-Verfahren gibt die Anzahl der im E/A-Durchführungsport eingereichten Fadenbetriebe zurück.

Arbeitsschlangen-Zustandklasse

[0088] Die Arbeitsschlangen-Zustandklasse verwaltet den Zustand der Arbeitsschlange **12**.

[0089] Das Besetzt-Faden-Zählwert-Attribut ist ein gezähltes Ereignis, das signalisiert, wann die Anzahl der besetzten Fäden 0 ist; d. h. wenn die Arbeitsschlange **12** 0 Fadenbetriebe hat.

[0090] Die Enter- und Exit-Zustand-Verfahren rufen abhängig vom aktuellen Zustand der Arbeitsschlange durch zu den entsprechenden Verfahren in den Klassen Arbeitsschlange-Gestoppt und Arbeitsschlange-Unterbrochen.

[0091] Die Start-Schlangen-, Stopp-Schlangen-, Unterbreche-Schlangen und Wiederaufnahme-Schlange-Verfahren rufen abhängig vom aktuellen Zustand der Arbeitsschlange durch zum passenden Verfahren der entsprechenden Verfahren in den Klassen Arbeitsschlangen-Gestoppt-Zustand, Arbeitsschlangen-Unterbreche-Zustand und Arbeitsschlangen-Laufend-Zustand.

[0092] Das Schlangen-Arbeitsverfahren ruft zurück zum Arbeitsschlangen-Zustand-Controller, um die Arbeit aus der Schlange zum E/A-Durchführungsport zu entnehmen.

[0093] Die Freigabe-Schlangen-Arbeitsfäden, Start-Arbeitsfäden und Schalte-Zustände-Verfahren rufen durch zu den entsprechenden Verfahren im Arbeitsschlangen-Zustand-Controller.

[0094] Das Wartezustand-Synchronisier-Verfahren wartet auf das Besetzt-Faden-Gezählt-Ereignis, um zu signalisieren, dass die Arbeitsschlange im Wartezustand ist, und kehrt dann zurück.

[0095] Das Inkrementier-Besetzt-Faden-Verfahren inkrementiert die Anzahl der Besetzt-Fäden im aktuellen Zustand, und das Senke-Besetzt-Faden-Verfahren senkt die Zahl der Besetzt-Fäden im aktuellen Zustand.

Arbeitsschlange-Gestoppt-Zustand-Klasse

[0096] Die Arbeitsschlange-gestoppt-Zustand-Klasse schließt Attribute und Verfahren ein, die das Verhalten der Arbeitsschlange im gestoppten Zustand implementieren.

[0097] Das Kein-Betrieb-Attribut ist ein Fadenbetrieb, der keine Arbeit durchführt, die für die Fadenausführung rückgeführt wird, wenn im gestoppten Zustand aus der Schlange herausgenommen wird.

[0098] Der Faden-Zählwert ist die Anzahl der Fäden **22** im Faden-Pool **14**.

[0099] Das Enter-Zustand-Verfahren gibt alle Fäden zur Beendigung frei, und das Exit-Zustand-Verfahren inkrementiert den Faden-Besetzt-Zählwert für jeden Faden im Faden-Pool und ruft das Verfahren

Start-Arbeitsfäden auf.

[0100] Das Start-Schlange-Verfahren ändert die Arbeitsschlange auf den unterbrochenen Zustand, und das Unterbrochen-Schlange-Verfahren ruft das Start-Schlange-Verfahren mit demselben Ergebnis auf.

[0101] Das Schlangenarbeitsverfahren tut nichts, da die Arbeit nicht im gestoppten Zustand in die Schlange eingereicht werden kann.

[0102] Das Ausreihungs-Arbeitsverfahren ruft den Arbeitsschlangen-Zustand-Controller auf, um vom E/A-Durchführungsport einen Fadenbetrieb zu erhalten. Wenn der Fadenbetrieb gültig ist, wird stattdessen ein Kein-Betrieb rückgeführt. Dies erlaubt der Arbeitsschlange, im gestoppten Zustand zu entleeren. Wenn die rückgeführte Arbeit NULL (ein Wert 0) ist, wird der Besetzt-Faden-Zählwert gesenkt und NULL rückgeführt.

Arbeitsschlangen-Unterbrochen-Zustand-Klasse

[0103] Der Arbeitsschlangen-Unterbrochen-Zustand implementiert das Verhalten der Arbeitsschlange im unterbrochenen Zustand.

[0104] Das Faden-Unterbrecher-Attribut ist ein Ereignis, das gesetzt wird, wenn Fäden ablaufen dürfen, und es wird gelöscht, wenn Fäden unterbrochen werden sollen.

[0105] Das Kein-Betrieb-Attribut ist ein Fadenbetrieb, der nichts tut, und das vom Ausreihungs-Arbeitsverfahren rückgegeben wird, nachdem der Faden-Unterbrecher signalisiert wird.

[0106] Das Enter-Zustand-Verfahren setzt das Faden-Unterbrecher-Ereignis zurück, um das Ausreihen der Arbeit zu sperren und alle Fäden mit einem Kein-Betrieb freizugeben. Dies erzwingt, dass die Fäden herausgenommen werden und in den unterbrochenen Zustand eintreten und nach dem Faden-Unterbrecher-Ereignis sperren.

[0107] Das Exit-Zustand-Verfahren setzt das Faden-Unterbrecher-Ereignis, wodurch erlaubt wird, dass die Ausreihungsbetriebe fortfahren.

[0108] Das Stopp-Schlange-Verfahren ändert die Arbeitsschlange auf den gestoppten Zustand.

[0109] Das Wiederaufnahme-Schlange-Verfahren ändert die Arbeitsschlange auf den laufenden Zustand.

[0110] Das Ausreihungs-Arbeitsverfahren senkt den Besetzt-Faden-Zählwert. Der Faden ist nicht belegt, wenn er auf das Signal des Faden-Unterbrechers wartet. Das Ausreihungs-Arbeitsverfahren wartet dann darauf, dass der Faden-Unterbrecher signalisiert wird, woraufhin der Besetzt-Faden-Zählwert inkrementiert wird und ein Kein-Betrieb rückgeführt wird. Der Faden führt dann den Kein-Betrieb aus und wartet auf den nächsten verfügbaren Betrieb.

Arbeitsschlangen-Laufend-Zustand-Klasse

[0111] Die Arbeitsschlangen-Laufend-Zu-

stand-Klasse implementiert das Verhalten der Arbeitsschlange im laufenden Zustand.

[0112] Das Stopp-Schlange-Verfahren ändert den Zustand der Arbeitsschlange von laufend auf gestoppt, und das Unterbreche-Schlange-Verfahren ändert den Zustand der Arbeitsschlange von laufend auf unterbrochen.

[0113] Das Ausreihungs-Arbeitsverfahren senkt als erstes den Besetzt-Faden-Zählwert., da ein Faden nicht belegt ist, wenn er auf Arbeit von der Arbeitsschlange wartet. Das Verfahren erhält dann den nächsten Fadenbetrieb durch das Aufrufen der Arbeitsschlangen-Implementierung. Der Besetzt-Faden-Zählwert wird dann inkrementiert und ein Zeiger wird zum herausgenommenen Fadenbetrieb zurückgeführt.

[0114] Alle relevanten Attribute und Verfahren, die Glieder verschiedener Klassen sind, werden weiter in den Objekt-Interaktionsdiagrammen beschrieben, die folgen.

[0115] **Fig. 4** ist ein Objekt-Interaktionsdiagramm, das das Einreihen in die Schlange eines Fadenbetriebs darstellt, wenn sich der Faden-Pool in einem laufenden oder unterbrochenen Zustand befindet. In diesem Objekt-Interaktionsdiagramm sowie in anderen, die folgen, wurden viele der "Rückkehr"-Steuerlinien weggelassen, um die Überfüllung in den Zeichnungen zu reduzieren. Es wird jedoch geschätzt sein, dass Rückführungen, selbst wenn sie nicht gezeigt werden, einbegriffen sind.

[0116] Die Start- und Endknoten stellen ein Aufrufprogramm, z. B. den Faden-Pool-Manager **18** (**Fig. 1**), dar. Um einen Fadenbetrieb **20** in die Schlange einzureihen, wird das Schlangen-Arbeitsverfahren des Arbeitsschlangen-Objekts mit Bezug auf den einzureihenden Fadenbetrieb aufgerufen, der durch zum Schlangen-Arbeitsverfahren des Arbeitsschlangen-Zustand-Controller-Objekts aufgerufen wird.

[0117] Der Arbeitsschlangen-Zustand-Controller ruft sein Erhalte-Aktuelles-Arbeitsschlangen-Zustand-Verfahren auf, um den aktuellen Zustand des Faden-Pools **14** zu erhalten. Das Erhalte-Aktuelle-Arbeitsschlangen-Zustand-Verfahren ruft das Warte-Verfahren des Betriebs-Synchr.-Objekt dazu auf, mit anderen Fäden zu synchronisieren. Das bedeutet, dass, wenn ein anderer Faden in einem kritischen Abschnitt ausgeführt wird (z. B. indem er den Zustand des Faden-Pools ändert), das Warte-Verfahren den Aufruffaden dazu veranlasst zu warten, bis der andere Faden seinen kritischen Abschnitt ausgeführt hat, und die Steuerung wird dann zum Warte-Schlangen-Zustand-Controller zurückgeführt. Das Erhalte-Aktuelles-Arbeitsschlangen-Zustand-Objekt erhält dann einen Zeiger für das aktuelle Zustand-Objekt (für dieses Diagramm ein Zeiger für ein Arbeitsschlangen-Laufend-Zustand-Objekt) und löst dann die Synchronisation aus. Das Erhalte-Aktuelles-Arbeitsschlangen-Zustand-Verfahren gibt dann den Zeiger an das Schlangen-Arbeitsverfahren des Ar-

beit-Zustand-Controller-Objekts zurück.

[0118] Mit dem Zeiger für das Arbeitsschlangen-Laufend-Zustand-Objekt ruft der Arbeitsschlangen-Zustand-Controller das Schlangenarbeitsverfahren auf, das den Arbeitsschlangen-Controller-Bezug ("Bezug" bedeutet in diesem Fall Zeiger) sichert. Das Arbeitsschlangen-Laufend-Zustand-Objekt ruft dann das Abwickle-Schlangen-Arbeitsverfahren zurück in den Arbeitsschlangen-Zustand-Controller, der das Schlangenarbeitsverfahren im Arbeitsschlangen-Implementierung-Objekt aufruft. Das Arbeitsschlangen-Implementierung-Objekt inkrementiert die Schlangentiefe und gibt die Steuerung an den Arbeitsschlangen-Zustand-Controller zurück. Die Steuerung wird schließlich zum ursprünglichen aufrufenden Programm zurückgeführt.

[0119] **Fig. 5** ist ein Objekt-Interaktionsdiagramm, das den Übergang des Faden-Pools von einem laufenden in einen unterbrochenen Zustand darstellt. In einer Verwendungsform der Erfindung steuert der Faden-Pool-Manager **18** den Zustand des Faden-Pools **14**. Der Faden-Pool-Manager **18** ruft das Unterbreche-Schlange-Verfahren des Arbeitsschlangen-Objekts auf, um die Zustandsänderung von laufend auf unterbrochen zu starten, die zum Unterbreche-Schlange-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts durchruft.

[0120] Das Arbeitsschlangen-Zustand-Controller-Objekt erhält dann mittels Verwendung des Erhalte-aktuellen-Arbeitsschlangen-Zustands, wie oben beschrieben, den aktuellen Zustand des Faden-Pools **14**. Der Zeiger für das Arbeitsschlangen-Laufend-Zustand-Objekt wird dann verwendet, um das Unterbreche-Schlange-Verfahren im Objekt aufzurufen. Das Arbeitsschlangen-Laufend-Zustand-Objekt ruft zurück zum Schalte-Zustände-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts, das die Fadensynchronisierung mit dem Warte-Verfahren aufruft, verlässt den aktuellen Zustand und zeigt den neuen Zustand an. Das bedeutet, dass der Zeiger auf das Arbeitsschlangen-Unterbrochen-Zustand-Objekt geschaltet wird. Wenn ein Zustand-Übergang erfolgt, haben sowohl der alte Zustand als auch der neue Zustand eine Chance, um irgendeine Austritts- oder Eintrittstätigkeit durchzuführen. Das "Verlasse Zustand ()"-Verfahren wird vom alten Zustand für die Austrittstätigkeiten verwendet.

[0121] Das Schalte-Zustand-Verfahren ruft dann das Enter-Zustand-Verfahren des Arbeitsschlangen-Unterbrochen-Zustand-Objekts auf. Das Enter-Zustand-Verfahren setzt das Unterbrecher-Ereignis zurück. Mit anderen Worten wird die Synchronisation im Unterbrecher-Ereignis freigegeben, das im wesentlichen irgendwelche unterbrochenen Fäden freigibt, die auf das "Unterbrecher-Ereignis" warten. Dann ruft das Enter-Zustand-Verfahren mit einem NOP-(Kein-Betrieb)-Parameter zurück zum Freigabe-Schlangen-Eingereiht-Arbeitsfäden-Verfahren, um anzuzeigen, dass ein Kein-Betrieb für jeden Faden in die Schlange eingereiht wird. Das Schlangen-

arbeitsverfahren des Arbeitsschlangen-Implementierung-Objekt wird mit einem Kein-Betrieb-Parameter für jeden der Fäden **22** im Faden-Pool **14** aufgerufen. Jeder Faden wird schließlich einen Kein-Betrieb aus der Schlange ausreihen, was bewirken wird, dass der Faden in den unterbrochenen Zustand tritt. Die Steuerung wird dann zum Arbeitsschlangen-Zustand-Controller-Objekt zurückgeführt, der die Fadensynchronisierung auslöst und die Steuerung zum aufrufenden Programm zurückführt.

[0122] **Fig. 6** ist ein Objekt-Interaktionsprogramm, das den Übergang des Faden-Pools von einem laufenden auf einen gestoppten Zustand darstellt. Der Faden-Pool-Manager **18** ruft das Stopp-Schlange-Verfahren des Arbeitsschlangen-Objekts auf, um die Zustandsänderung von laufend auf gestoppt zu starten, die durch zum Stopp-Schlange-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts aufruft.

[0123] Das Arbeitsschlangen-Zustand-Controller-Objekt erhält dann mittels Verwendung des Erhalte-Aktuellen-Arbeitsschlangen-Zustand-Objekts den aktuellen Zustand des Faden-Pools **14**. Der Zeiger für das Arbeitsschlangen-Laufend-Zustand-Objekt wird dann verwendet, um das Stopp-Schlange-Verfahren im Objekt aufzurufen. Das Arbeitsschlangen-Laufend-Zustand-Objekt ruft zurück zum Schalte-Zustände-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts mit dem ARBEITS_WARTESCHLANGEN_GESTOPPT-Parameter, das die Fadensynchronisierung mit dem Warteverfahren aufruft, verlässt den aktuellen Zustand und zeigt den neuen Zustand an. Das bedeutet, dass der Zeiger auf das Arbeitsschlangen-Gestoppt-Zustand-Objekt geschaltet wird.

[0124] Das Schalte-Zustand-Verfahren ruft dann das Enter-Zustand-Verfahren des Arbeitsschlangen-Gestoppt-Zustand-Objekts auf. Das Enter-Zustand-Verfahren bildet das Gegenstück des Verlasse-Zustand-Verfahrens. Es wird aufgerufen, um irgendwelche Eintrittstätigkeiten im neuen Zustand durchzuführen, wenn ein Zustand-Übergang stattfindet. Dann ruft das Enter-Zustand-Verfahren mit einem NULL-Parameter zurück zum Freigabe-Schlangen-Arbeitsfäden-Verfahren, um anzuzeigen, dass für jeden Faden eine Null Schlangeneingereiht werden muss. Das Schlangen-Null-Arbeitsverfahren des Arbeitsschlangen-Implementierung-Objekts wird für jeden der Fäden **22** im Faden-Pool **14** aufgerufen. Jeder Faden wird schließlich einen NULL-Betrieb ausreihen, was bewirken wird, dass der Faden in den gestoppten Zustand eintritt. Die Steuerung wird dann zum Arbeitsschlangen-Zustand-Controller-Objekt rückgeführt, das die Fadensynchronisierung freigibt und die Steuerung zum aufrufenden Programm zurückgibt.

[0125] **Fig. 7** ist ein Objekt-Interaktionsdiagramm, das den Übergang des Faden-Pools von einem unterbrochenen auf einen laufenden Zustand darstellt. Der Faden-Pool-Manager **18** ruft das Wiederaufnah-

me-Schlange-Verfahren des Arbeitsschlangen-Objekts auf, um die Zustandsänderung von unterbrochen auf laufend zu beginnen, die durch zum Wiederaufnahme-Schlange-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts aufruft.

[0126] Das Arbeitsschlangen-Zustand-Controller-Objekt erhält dann den aktuellen Zustand des Faden-Pools **14**, indem wie oben beschrieben der Erhalte-Aktuellen-Arbeitsschlangen-Zustand verwendet wird. Der Zeiger für das Arbeitsschlangen-Unterbrochen-Zustand-Objekt wird dann verwendet, um das Wiederaufnahme-Schlange-Verfahren im Objekt aufzurufen. Das Arbeitsschlangen-Unterbrochen-Zustand-Objekt ruft zurück zum Schalte-Zustände-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts mit dem ARBEITS_WARTESCHLANGEN_GESTOPPT-Parameter, das die Fadensynchronisierung mit dem Warteverfahren aufruft, den aktuellen Zustand verlässt und auf den neuen Zustand verweist. Der Zeiger wird auf das Arbeitsschlangen-Laufend-Zustand-Objekt geschaltet. Das Verlasse-Zustand-Verfahren ruft das Setze-Verfahren des Faden-Unterbrecher-Objekts auf, um die Zustand-Übergänge in eine serielle Form zu bringen, damit verhindert wird, dass die Fäden untereinander die Zustände ändern. Der gesetzte Aufruf wird die Synchronisierung derart erhöhen, dass der Faden unterbrochen wird, wenn irgendein anderer Faden bereits als gesetzt aufgerufen ist.

[0127] Das Schalte-Zustand-Verfahren ruft dann das Enter-Zustand-Verfahren des Arbeitsschlangen-Laufend-Zustand-Objekts auf. Die Steuerung wird daraufhin zum Arbeitsschlangen-Zustand-Controller-Objekt zurückgeführt, das die Fadensynchronisierung auslöst und die Steuerung zum ausführenden Programm zurückgibt.

[0128] **Fig. 8** ist ein Objekt-Interaktionsprogramm, das den Übergang des Faden-Pools von einem unterbrochenen auf einen gestoppten Zustand darstellt. Das Verfahren ist ähnlich dem Übergang vom laufenden Zustand auf dem gestoppten Zustand, mit der Ausnahme, dass das Verlasse-Zustand-Verfahren des Arbeitsschlangen-Unterbreche-Zustand-Objekts das Setze-Verfahren des Faden-Unterbreche-Vorrichtung-Objekts aufruft.

[0129] **Fig. 9** ist ein Objekt-Interaktionsprogramm, das den Übergang des Faden-Pools von einem gestoppten auf einen unterbrochenen Zustand darstellt. Der Faden-Pool-Manager **18** ruft das Start-Schlange-Verfahren des Arbeitsschlangen-Objekts auf, um die Zustandsänderung von gestoppt auf unterbrochen zu starten, die durch zum Start-Schlange-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts aufruft.

[0130] Das Arbeitsschlangen-Zustand-Controller-Objekt erhält dann mittels Verwendung des Erhalte-Aktuellen-Arbeitsschlangen-Zustands den aktuellen Zustand des Faden-Pools **14**. Der Zeiger für das Arbeitsschlangen-Gestoppt-Zustand-Objekt wird dann verwendet, um das Start-Schlange-Verfahren

im Objekt aufzurufen. Das Arbeitsschlangen-Gestoppt-Zustand-Objekt ruft zurück zum Schalte-Zustände-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts mit dem ARBEITS_WARTESCHLANGEN_UNTERBROCHE N-Parameter, wobei die Fadensynchronisierung mit dem Warteverfahren aufgerufen wird.

[0131] Das Verlasse-Zustand-Verfahren ruft dann das Arbeitsschlangen-Gestoppt-Zustand-Objekts auf, das zurück zum Start-Arbeitsfäden-Verfahren des Arbeitsschlangen-Zustand-Controller-Objekts ruft. Das Start-Arbeitsfäden-Verfahren wartet zunächst darauf, dass irgendwelche laufenden Fäden ausgeführt werden (und aus dem Faden-Pool gelöscht werden) und ruft dann das Start-Fäden-Verfahren des Arbeitsschlangen-Faden-Pool-Objekts auf, um die neuen Fäden im Faden-Pool 14 zu starten. Nachdem die Fäden gestartet sind, wird die Steuerung zum Arbeitsschlangen-Zustand-Controller-Objekt zurückgegeben, und die Verarbeitung fährt wie oben in **Fig. 5** beschrieben fort.

[0132] **Fig. 10** ist ein Objekt-Interaktionsdiagramm, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem unterbrochenen Zustand befindet. Die Fadenausführung wird mit dem Betriebs-Hauptverfahren des Arbeitsschlangen-Zustand-Controller-Objekts gestartet. Der Faden erhält zuerst den aktuellen Zustand des Faden-Pools, wie oben beschrieben, und ruft dann das Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-Unterbrochen-Zustand-Objekts auf.

[0133] Das Ausreihungs-Arbeitsverfahren senkt als erstes der Zählwert der belegten Fäden. Da für den Faden noch keine Arbeit herausgenommen wurde, ist der Faden nicht besetzt. Dann wird das Warteverfahren des Faden-Unterbrecher-Objekts aufgerufen, um darauf zu warten, signalisiert zu werden. Wenn die Steuerung zum Arbeitsschlangen-Unterbrochen-Zustand-Objekt zurückgeführt wird, wird der Zählwert der besetzten Fäden inkrementiert, und ein Kein-Betrieb wird als Arbeit für den Faden zurückgeführt. Das Arbeitsschlangen-Zustand-Controller-Objekt ruft dann das Betrieb-Hauptverfahren des Kein-Betrieb-Objekts auf. Das Kein-Betrieb-Objekt ist ein Fadenbetrieb. Der Faden führt den Kein-Betrieb-Fadenbetrieb aus und wartet dann auf den nächsten verfügbaren Betrieb. Unter einem normalen Betrieb wird die NOP-Betriebsausführung ausgeführt und der Faden wird realisieren, dass sich der Zustand geändert hat. Der Grund für den NOP liegt darin, irgendwelche unterbrochenen Fäden aufzuwecken, so dass sie realisieren können, dass sich der Zustand geändert hat.

[0134] **Fig. 11** ist ein Objekt-Interaktionsdiagramm, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem laufenden Zustand befindet. Die Fadenausführung wird mit dem Betriebs-Hauptverfahren des Arbeitsschlangen-Zustand-Controller-Objekts gestartet. Der Faden erhält zuerst den aktuellen Zustand des Faden-Pools, wie oben beschrie-

ben, und ruft dann das Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-Laufend-Zustand-Objekts auf. [0135] Das Ausreihungs-Arbeitsverfahren senkt den Zählwert der besetzten Fäden und ruft das Abwickle-Ausreihungs-Arbeitsverfahren des Schlangenzustand-Controller-Objekts auf, das das Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-Implementierung-Objekts aufruft. Das Arbeitsschlangen-Implementierung-Objekt erhält einen Fadenbetrieb von der Arbeitsschlange 12 (falls einer verfügbar ist), senkt die Tiefe der Arbeitsschlange und wirft den Fadenbetrieb an das Arbeitsschlangen-Zustand-Controller-Objekt heraus und gibt es daran zurück. Die Steuerung wird dann zum Arbeitsschlangen-Laufend-Zustand-Objekt und danach zum Arbeitsschlangen-Zustand-Controller-Objekt zurückgeführt. Das Betrieb-Hauptverfahren des Betriebsobjekts wird aufgerufen, um den Fadenbetrieb auszuführen. Das Betriebsobjekt wird aus einer vom Benutzer bestimmten Klasse instantiiert, die eine Fadenbetrieb-Klasse darstellt.

[0136] Wenn der Faden die Ausführung des Fadenbetriebobjekts durchgeführt hat, kehrt der Faden zurück, um den Zustand des Faden-Pools zu erhalten und einen anderen Fadenbetrieb aus der Schlange zu nehmen.

[0137] **Fig. 12** ist ein Objekt-Interaktionsdiagramm, das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem gestoppten Zustand befindet und die Arbeitsschlange nicht leer ist. Die Fadenausführung wird mit dem Betriebs-Hauptverfahren des Arbeitsschlangen-Zustand-Controller-Objekts gestartet. Der Faden erhält zuerst den aktuellen Zustand des Faden-Pools, wie oben beschrieben, und ruft dann das Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-gestoppt-Zustand-Objekts auf.

[0138] Das Ausreihungs-Arbeitsverfahren ruft das Abwickle-Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-Zustand-Controller-Objekts auf, das das Ausreihungs-Arbeitsverfahren des Arbeitsschlangen-Implementierung-Objekts aufruft. Das Arbeitsschlangen-Implementierung-Objekt erhält einen Fadenbetrieb von der Arbeitsschlange 12 (vorausgesetzt, dass einer verfügbar ist), senkt die Tiefe der Arbeitsschlange und wirft den Fadenbetrieb heraus und gibt ihn an das Arbeitsschlangen-Zustand-Controller-Objekt zurück. Die Steuerung wird dann, zum Arbeitsschlangen-Gestoppt-Zustand-Objekt rückgeführt. Wenn der vom Arbeitsschlangen-Implementierung-Objekt rückgeführte Fadenbetrieb gültig ist, wird ein Kein-Betrieb an das Arbeitsschlangen-Zustand-Controller-Objekt zurückgeführt.

[0139] Das Betrieb-Hauptverfahren des Betriebsobjekts wird aufgerufen, um den Kein-Betrieb-Fadenbetrieb auszuführen. Wenn der Faden den Kein-Betrieb-Fadenbetrieb ausgeführt hat, kehrt der Faden zurück, um den Zustand des Faden-Pools zu erhalten und einen weiteren Fadenbetrieb aus der Schlange zu nehmen.

[0140] **Fig. 13** ist ein Objekt-Interaktionsdiagramm,

das die Ausführung eines Fadens darstellt, wenn sich der Faden-Pool in einem gestoppten Zustand befindet und die Arbeitsschlange leer ist. Die Objekt-Interaktion für eine leere Arbeitsschlange **12**, wenn sich der Faden-Pool **14** in einem gestoppten Zustand befindet, ähnelt der Objektinteraktion, wenn die Arbeitsschlange nicht leer ist. Jedoch liegt der Unterschied darin, dass das Arbeitsschlangen-Implementierungs-Objekt die NULL zum Arbeitsschlangen-Zustand-Controller-Objekt zurückführt, das die Senkung des Besetzt-Faden-Zählwerts bewirkt, und der Faden wird zerstört.

[0141] Entsprechend stellt die vorliegende Erfindung unter weiteren Aspekten ein Verfahren zum Verwalten eines Faden-Pools und eine von den Fäden durchzuführende Arbeitsschlange bereit. Weitere Aspekte und Ausführungsformen der vorliegenden Erfindung werden den Fachleuten auf dem Gebiet in Zusammenhang mit der Betrachtung der Beschreibung und der praktischen Anwendung der hierin offenbarten Erfindung ersichtlich. Es ist beabsichtigt, die Beschreibung und die dargestellten Ausführungsformen lediglich als Beispiele zu betrachten, wobei der wahre Schutzbereich der Erfindung durch die folgenden Ansprüche angezeigt wird.

Patentansprüche

1. Ein Computer-implementiertes Verfahren zum Verwalten eines Pools von Fäden für die Ausführung von Fadenbetrieben, worin der Pool (**14**) von Fäden instantiiert wird, damit er eine Mehrzahl von Fäden (**22**) hat, gekennzeichnet durch die Instantiierung einer Arbeitsschlange (**12**), die einen Programm-steuerbaren Zustand aufweist; die Einreihung der Fadenbetriebe in die Arbeitsschlange, um von Fäden im Faden-Pool ausgeführt zu werden, wenn sich die Arbeitsschlange in einem ersten vorbestimmten Zustand befindet; und die Verweigerung der Einreihung der Fadenbetriebe in die Arbeitsschlange, wenn sich die Arbeitsschlange in einem zweiten von einer Mehrzahl an Zuständen befindet.

2. Das Verfahren nach Anspruch 1, das weiterhin umfasst, dass jeder der Fäden im Faden-Pool folgende Schritte wiederholt:
das Herausnehmen eines Fadenbetriebs aus der Arbeitsschlange; und
das Ausführen des Fadenbetriebs.

3. Das Verfahren nach Anspruch 2, das weiterhin das Einreihen eines Fadenbetriebs aus der Arbeitsschlange umfasst, wenn sich die Arbeitsschlange in einem laufenden Zustand befindet.

4. Das Verfahren nach Anspruch 3, das weiterhin das Einreihen eines Fadenbetriebs in die Arbeitsschlange umfasst, wenn sich die Arbeitsschlange in einem unterbrochenen Zustand befindet.

5. Das Verfahren nach Anspruch 4, das weiterhin das Ablehnen einer Anforderung zum Einreihen in die Arbeitsschlange eines Fadenbetriebs umfasst, wenn sich die Arbeitsschlange in einem gestoppten Zustand befindet.

6. Das Verfahren nach Anspruch 5, das, wenn der Zustand der Arbeitsschlange von laufend auf unterbrochen geändert wird, die Durchführung des Schritts zum Einstellen des Herausnehmens aus der Arbeitsschlange der Fadenbetriebe umfasst.

7. Das Verfahren nach Anspruch 6, das, wenn sich die Arbeitsschlange in einem unterbrochenen Zustand befindet, weiterhin folgendes umfasst:
dass jeder der Fäden im Faden-Pool damit fortfährt, Anforderungen zu unterbreiten, damit Fadenbetriebe aus der Arbeitsschlange herausgenommen werden; und
dass die Kein-Betrieb-Fadenbetriebe als Reaktion auf die Schlangen-Herausnahme-Anforderungen an die Fäden zurückgegeben werden.

8. Das Verfahren nach Anspruch 5, das, wenn der Zustand der Arbeitsschlange von laufend auf gestoppt geändert wird, weiterhin das Durchführen des Schritts zum Herausnehmen nicht ausgeführter Fadenbetriebe aus der Arbeitsschlange umfasst.

9. Das Verfahren nach Anspruch 8, das weiterhin die Erlaubnis umfasst, dass in Ausführung befindliche Fadenbetriebe abgeschlossen werden.

10. Das Verfahren nach Anspruch 8, das, wenn sich die Arbeitsschlange in einem gestoppten Zustand befindet, weiterhin umfasst:
dass jeder der Fäden im Faden-Pool damit fortfährt, Anforderungen zu unterbreiten, damit Fadenbetriebe aus der Arbeitsschlange herausgenommen werden; und
dass die Kein-Betrieb-Fadenbetriebe als Reaktion auf die Schlangen-Herausnahme-Anforderungen an die Fäden zurückgegeben werden.

11. Das Verfahren nach Anspruch 5, das weiterhin folgendes umfasst:
das Starten eines oder mehrerer zeitweiliger Fäden mit dem Faden-Pool; und
das Entfernen der zeitweiligen Fäden aus dem Faden-Pool, wenn die Arbeitsschlange auf einen gestoppten Zustand geändert wird.

12. Ein Computer-implementiertes Verfahren zum Verwalten der Ausführung der Fadenbetriebe, worin eine Mehrzahl an Faden-Pools instantiiert wird, die jeweils eine verknüpfte Mehrzahl von Fäden einschließen, gekennzeichnet durch die Instantiierung einer Mehrzahl an Arbeitsschlangen, die jeweils einen Programm-steuerbaren Zustand haben und mit einem der Faden-Pools ver-

knüpft sind;
 die Einreihung der Fadenbetriebe in die Arbeitsschlangen, um von Fäden in den verknüpften Faden-Pools ausgeführt zu werden, worin die Fadenbetriebe in eine Arbeitsschlange eingereiht werden können, wenn sich die Arbeitsschlange in einem ersten vorbestimmten Zustand befindet; und
 die Verweigerung der Einreihung der Fadenbetriebe in eine Arbeitsschlange, wenn sich die Arbeitsschlange in einem zweiten Zustand der Mehrzahl an Zuständen befindet.

13. Das Verfahren nach Anspruch 12, das weiterhin umfasst, dass jeder der Fäden im Faden-Pool folgende Schritte wiederholt:
 das Herausnehmen eines Fadenbetriebs aus der verknüpften Arbeitsschlange; und
 das Ausführen des Fadenbetriebs.

14. Das Verfahren nach Anspruch 13, das weiterhin folgendes umfasst:
 das Einreihen eines Fadenbetriebs in eine Arbeitsschlange, wenn sich die Arbeitsschlange in einem laufenden Zustand und in einem unterbrochenen Zustand befindet; und
 das Ablehnen einer Anforderung zum Einreihen eines Fadenbetriebs in eine Arbeitsschlange, wenn sich die Arbeitsschlange in einem gestoppten Zustand befindet.

15. Das Verfahren nach Anspruch 14, das weiterhin folgendes umfasst:
 das Starten von einem oder mehreren zeitweiligen Fäden mit einem oder mehreren der Faden-Pools; und
 das Entfernen des eines oder der mehreren zeitweiligen Fäden aus einem Faden-Pool, wenn die mit dem Faden-Pool verknüpfte Arbeitsschlange auf einen gestoppten Zustand geändert wird.

16. Ein Gerät zum Verwalten eines Pools von Fäden für die Ausführung von Fadenbetrieben, das folgendes umfasst:
 ein Mittel zur Instantiierung eines Faden-Pools, der eine Mehrzahl von Fäden hat;
 ein Mittel zur Instantiierung einer Arbeitsschlange, die über einen Programm-steuerbaren Zustand verfügt;
 ein Mittel zum Einreihen der Fadenbetriebe in die Arbeitsschlange für die Ausführung durch die Fäden im Faden-Pool, wenn sich die Arbeitsschlange in einem ersten vorbestimmten Zustand befindet; und
 ein Mittel zum Verweigern der Einreihung der Fadenbetriebe in die Arbeitsschlange, wenn sich die Arbeitsschlange in einem zweiten der Mehrzahl an Zuständen befindet.

17. Ein System zum Verwalten eines Faden-Pools für die Ausführung der Fadenbetriebe, wobei das System ein Faden-Pool-Objekt einschließt,

das eine Mehrzahl an Faden-Objekten hat, gekennzeichnet durch
 ein Arbeitsschlangen-Objekt (**12**), das einen Programm-steuerbaren Zustand hat;
 einen Arbeitsschlangen-Controller (**16**), der mit dem Arbeitsschlangen-Objekt und dem Faden-Pool-Objekt verbunden ist, wobei der Arbeitsschlangen-Controller konfiguriert und angeordnet ist, um Fadenbetriebe für die Ausführung in das Arbeitsschlangen-Objekt einzureihen, wenn sich das Arbeitsschlangen-Objekt in einem ersten Zustand befindet, und um die Einreihung eines Fadenbetriebs in das Arbeitsschlangen-Objekt zu verweigern, wenn sich das Arbeitsschlangen-Objekt in einem zweiten Zustand befindet; und
 einen Faden-Pool-Manager (**18**), der mit dem Arbeitsschlangen-Controller verbunden wird, wobei der Faden-Pool-Manager konfiguriert und angeordnet ist, um selektiv die Zustände des Arbeitsschlangen-Objekts zu ändern.

18. Das System nach Anspruch 17, worin jedes der Faden-Objekte konfiguriert und angeordnet ist, um die Fadenbetriebe aus der Arbeitsschlange herauszunehmen und um die Fadenbetriebe auszuführen.

19. Das System nach Anspruch 18, worin die Fadenbetriebe in das Arbeitsschlangen-Objekt eingereiht werden können, wenn sich das Arbeitsschlangen-Objekt in einem laufenden Zustand befindet.

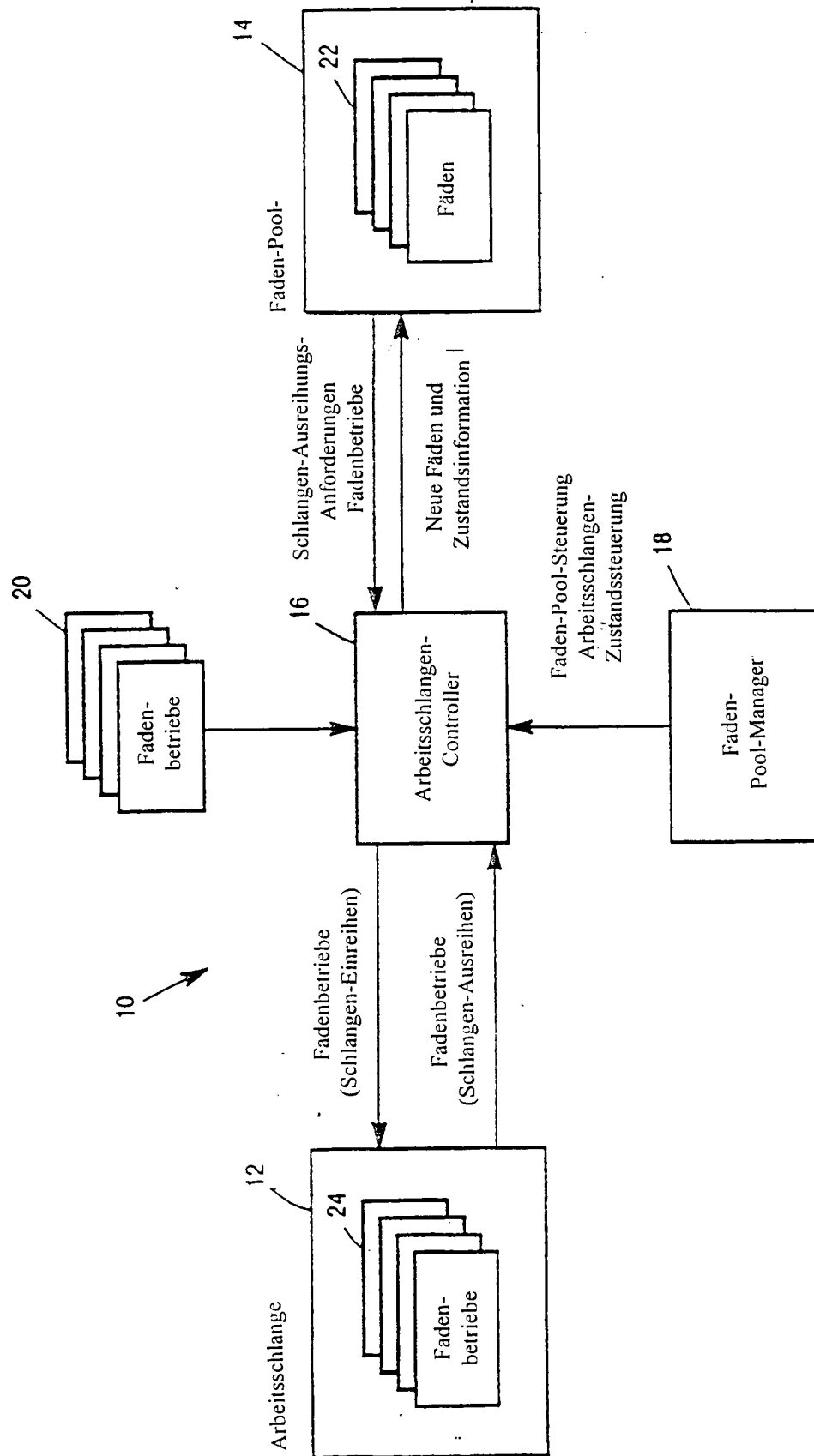
20. Das System nach Anspruch 19, worin die Fadenbetriebe in das Arbeitsschlangen-Objekt eingereiht werden können, wenn sich das Arbeitsschlangen-Objekt in einem unterbrochenen Zustand befindet.

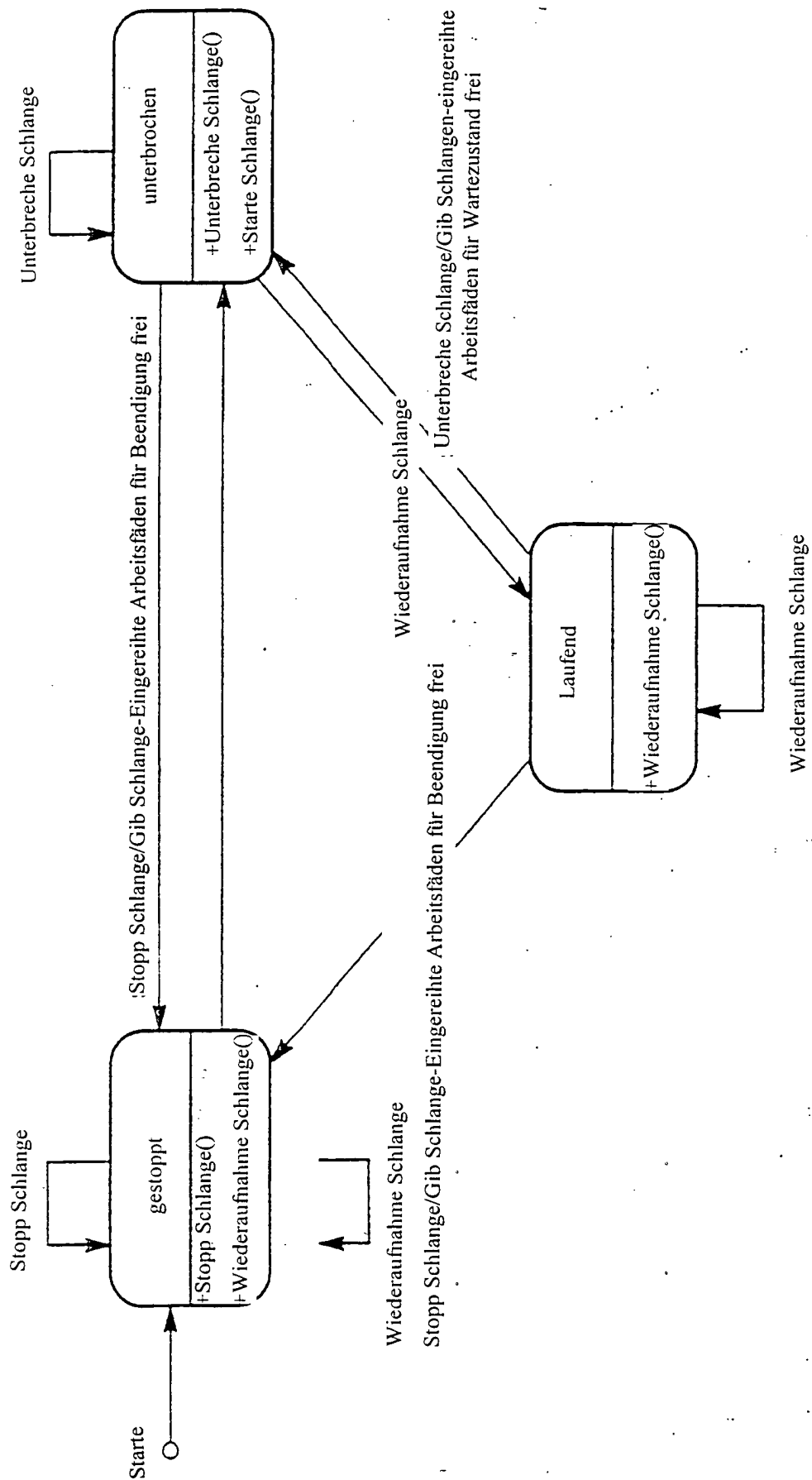
21. Das System nach Anspruch 20, worin Anforderungen zum Einreihen eines Fadenbetriebs in die Arbeitsschlange abgelehnt werden, wenn sich das Arbeitsschlangen-Objekt in einem gestoppten Zustand befindet.

22. Das System nach Anspruch 21, worin der Faden-Pool-Manager konfiguriert und angeordnet ist, um ein oder mehrere zeitweiligen Faden-Objekte mit dem Faden-Pool-Objekt zu starten und um die zeitweiligen Faden-Objekte aus dem Faden-Pool-Objekt zu entfernen, wenn das Arbeitsschlangen-Objekt in einen gestoppten Zustand geändert wird.

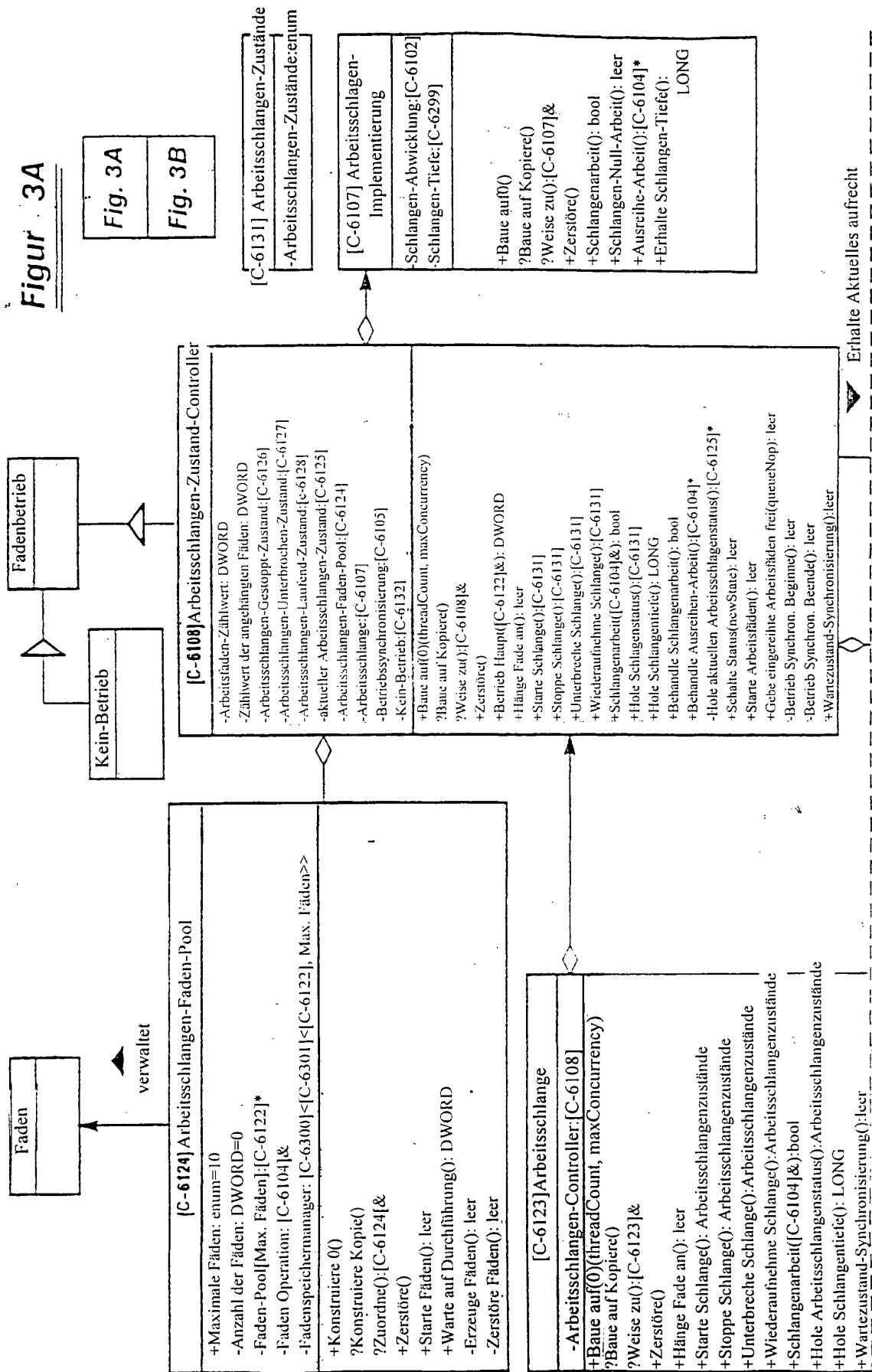
Es folgen 14 Blatt Zeichnungen

Anhängende Zeichnungen

**Figur 1**

**Figur 2**

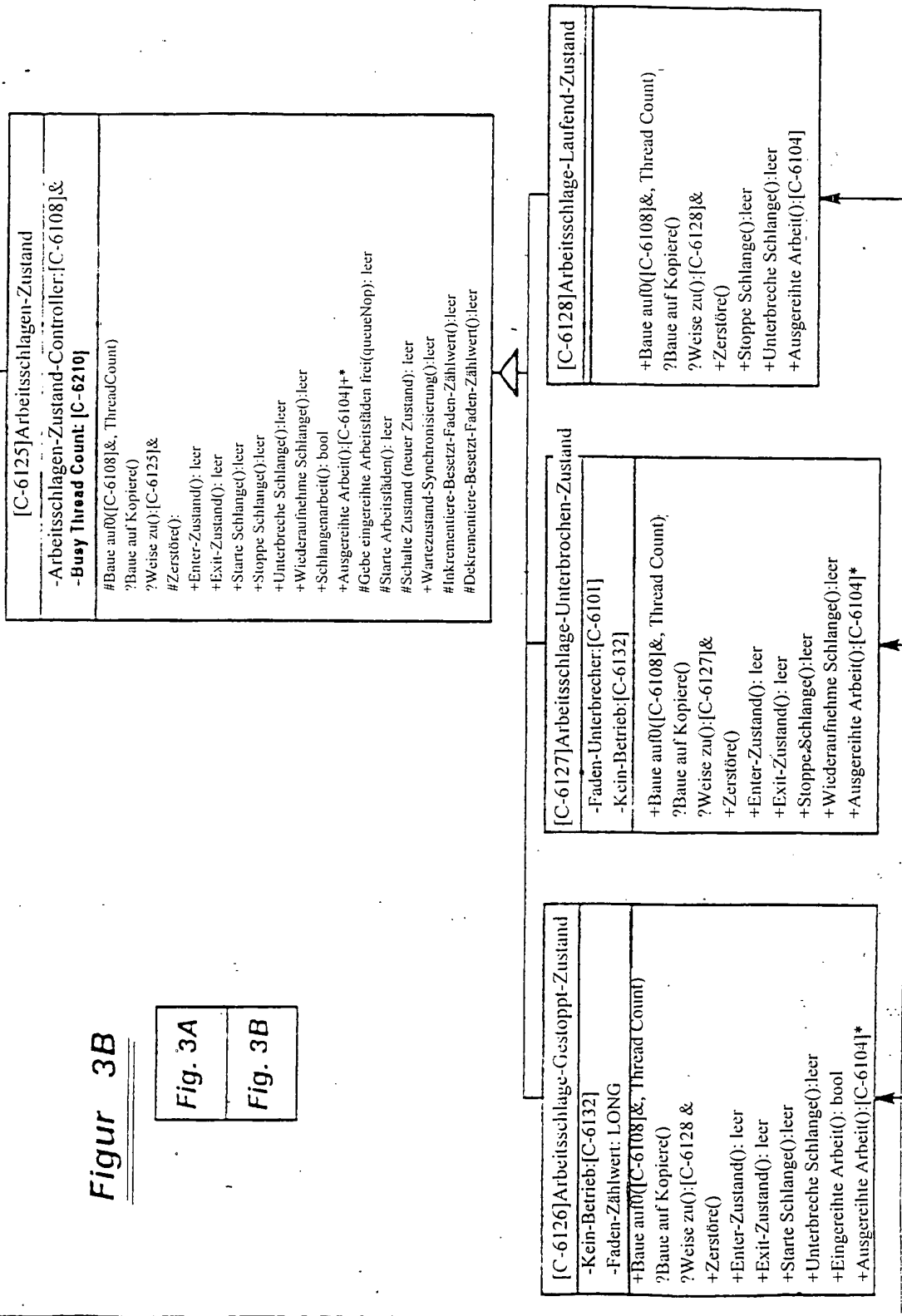
Figur 3A

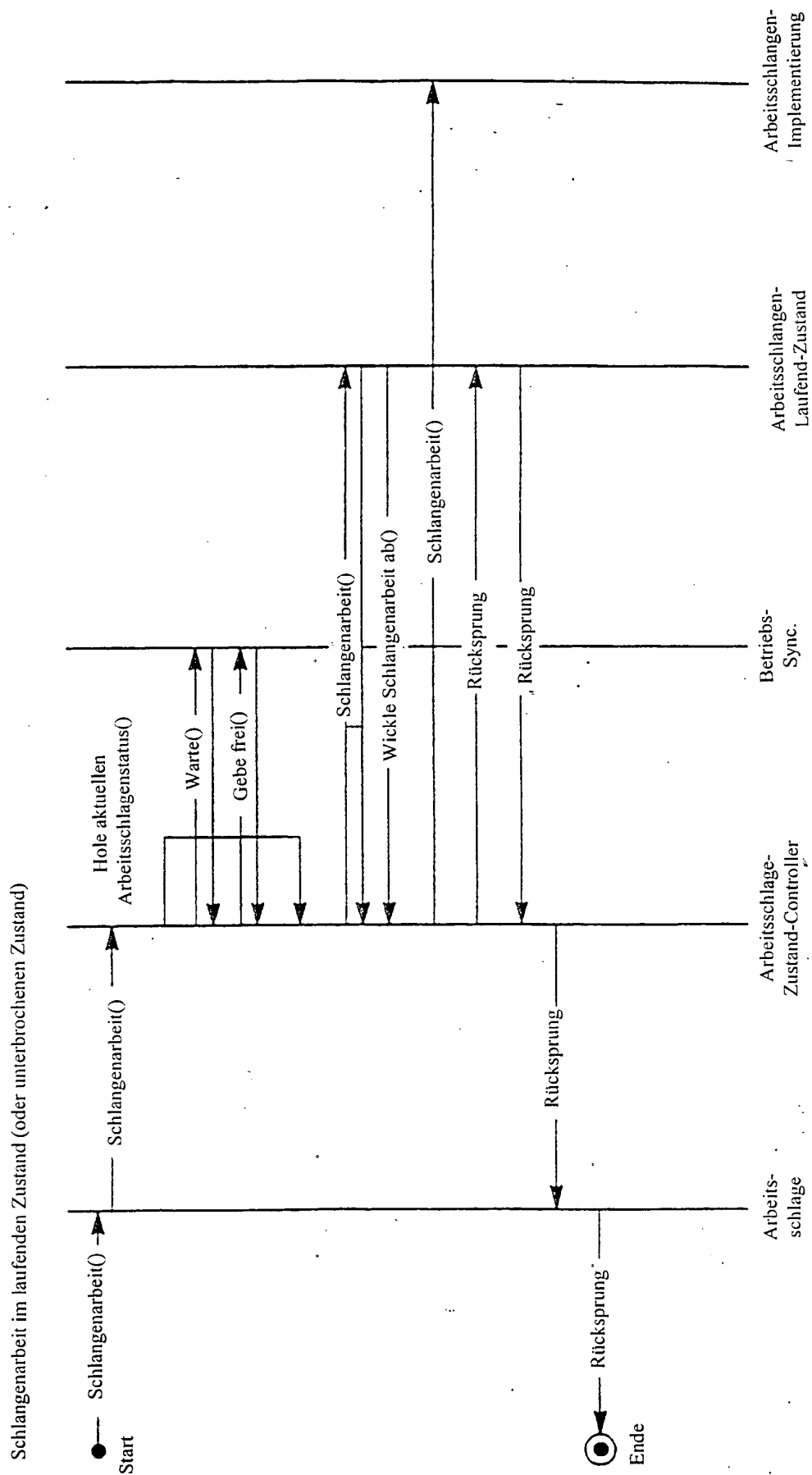


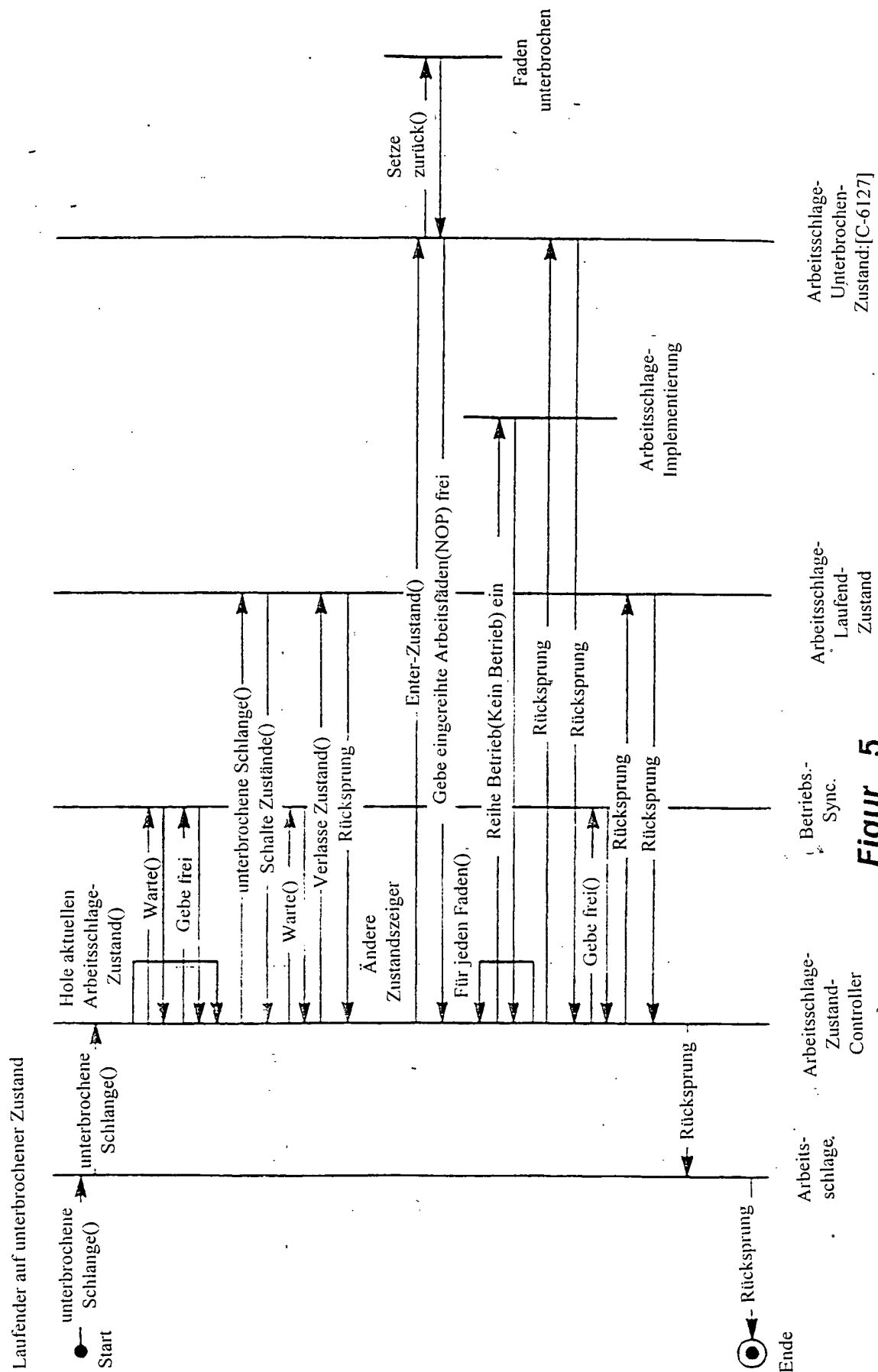
Figur 3B

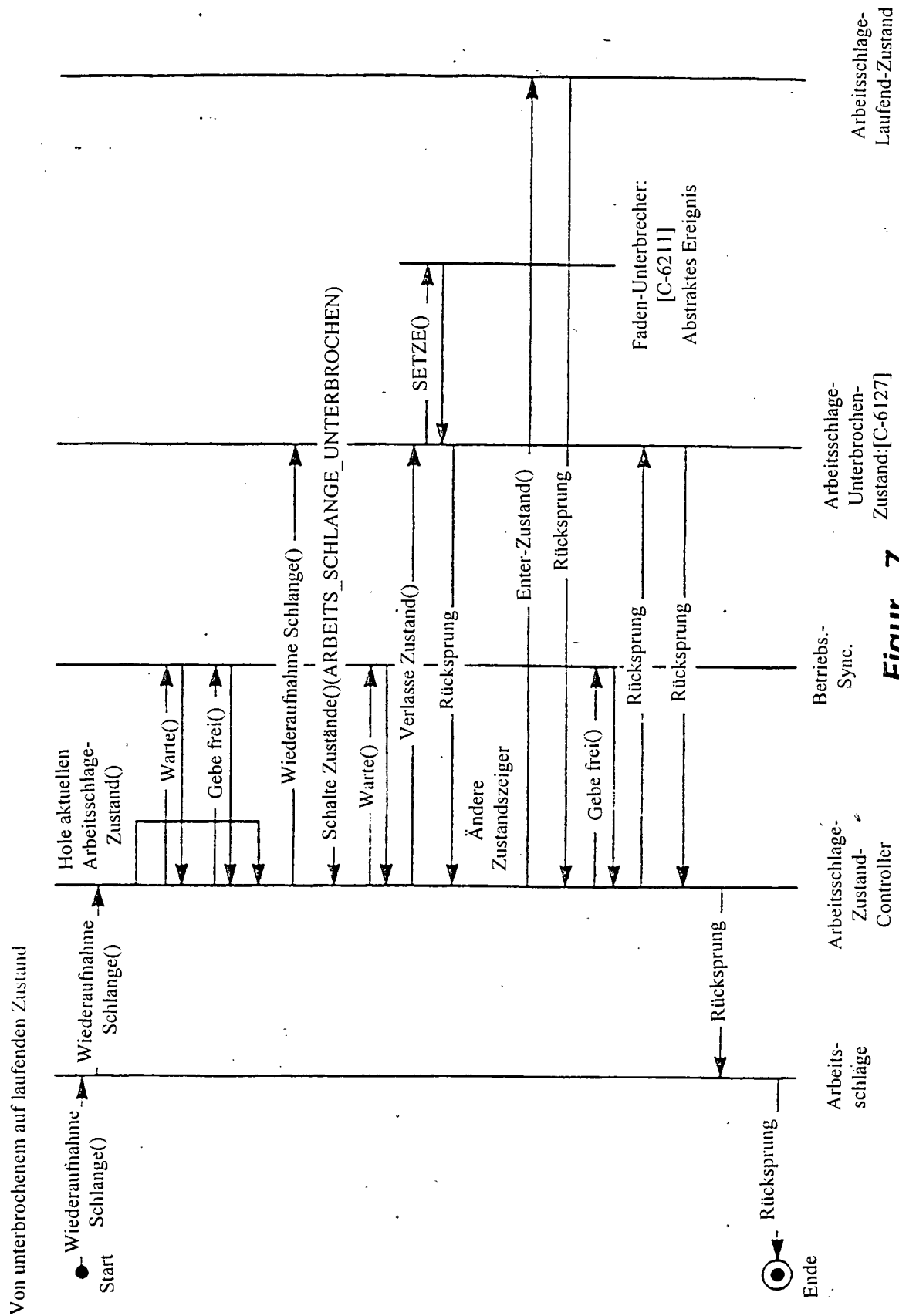
Fig. 3A

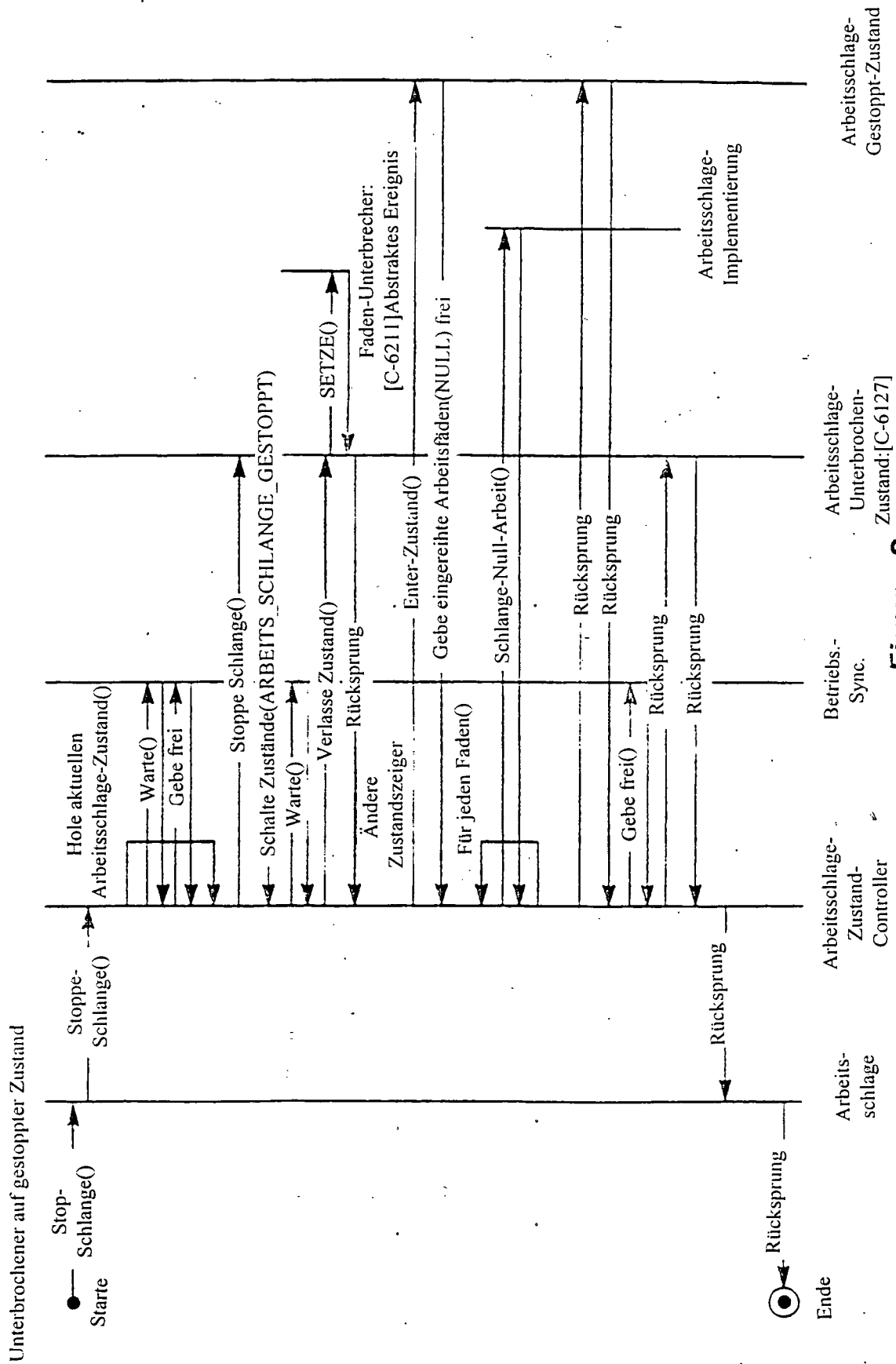
Fig. 3B

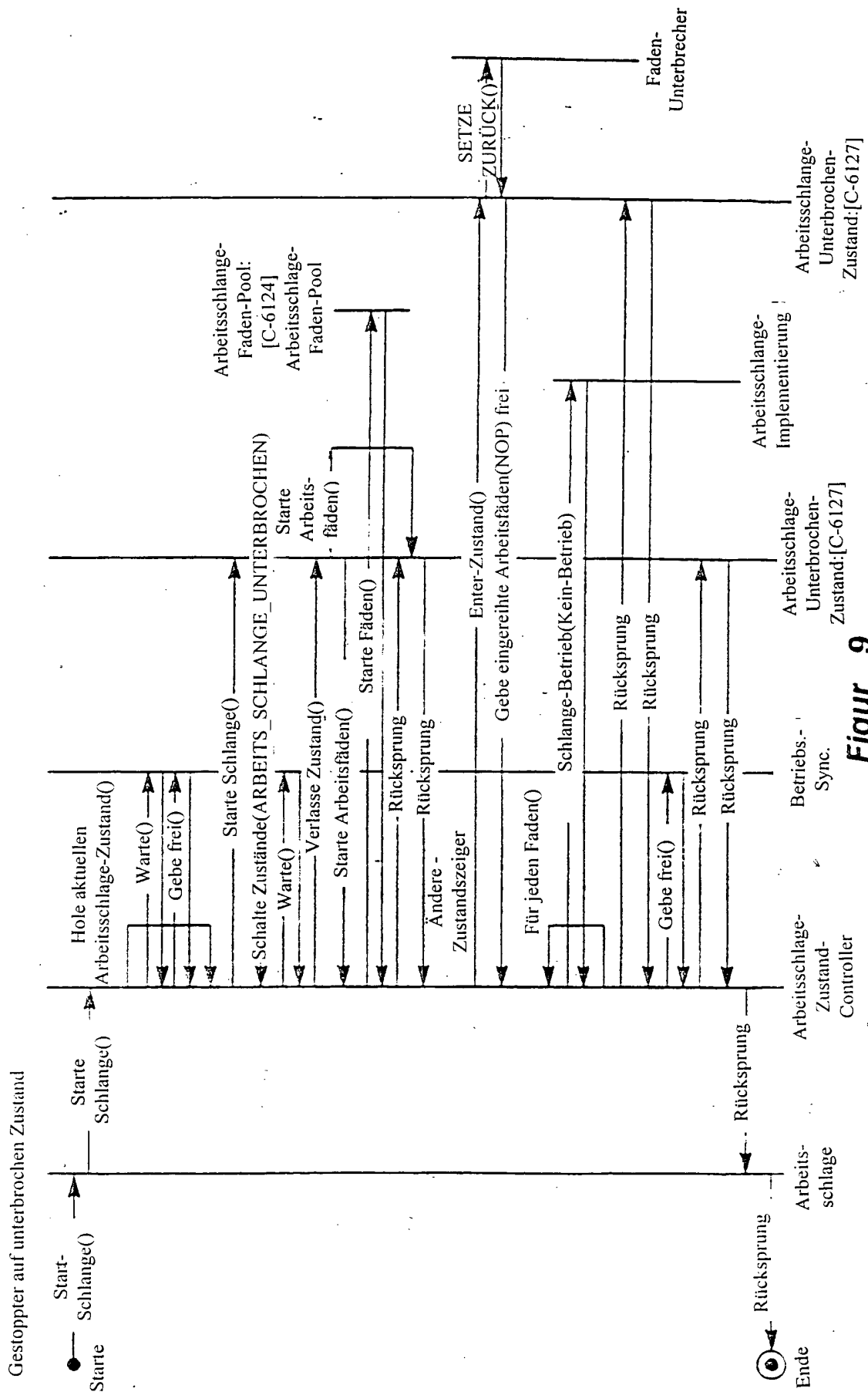


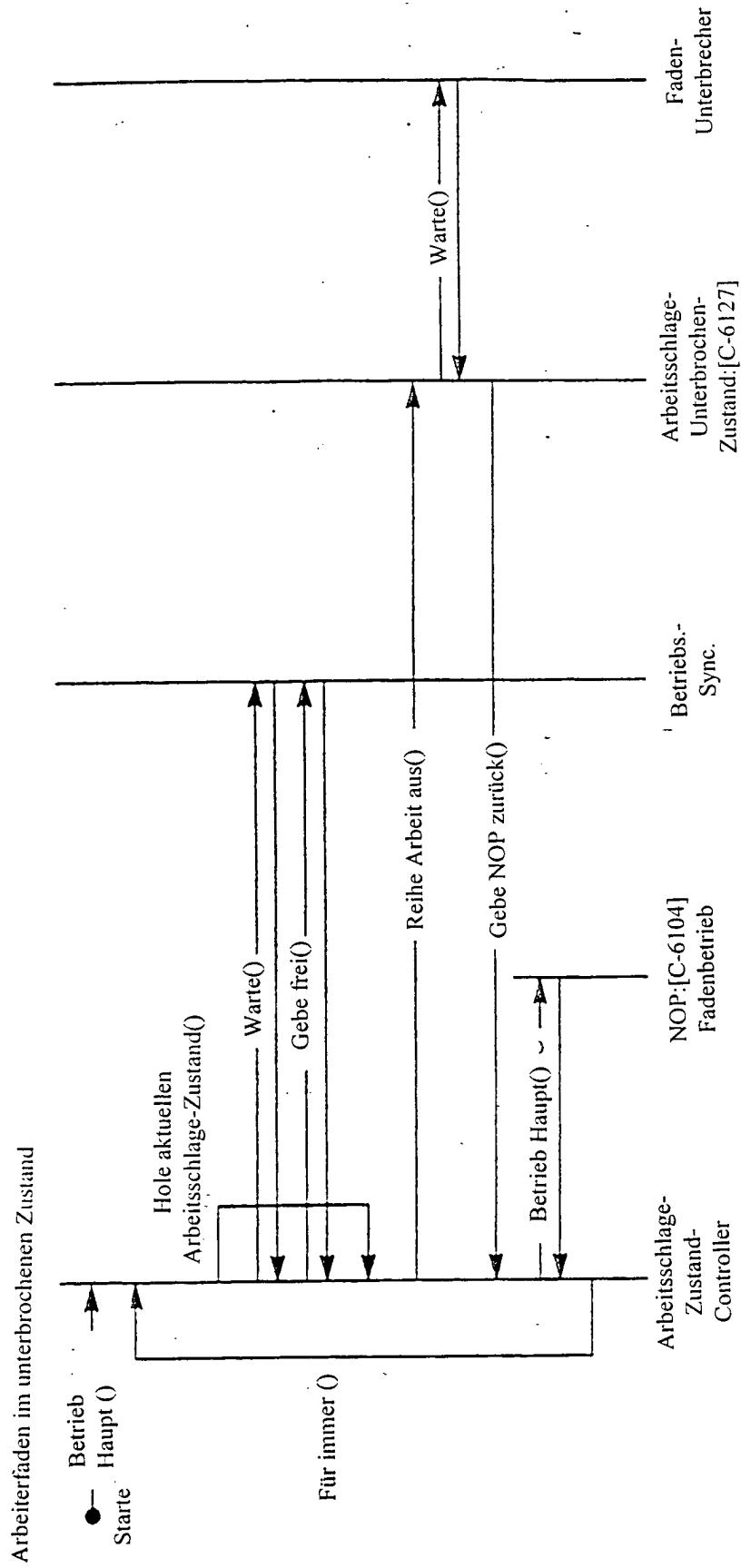
**Figur 4**

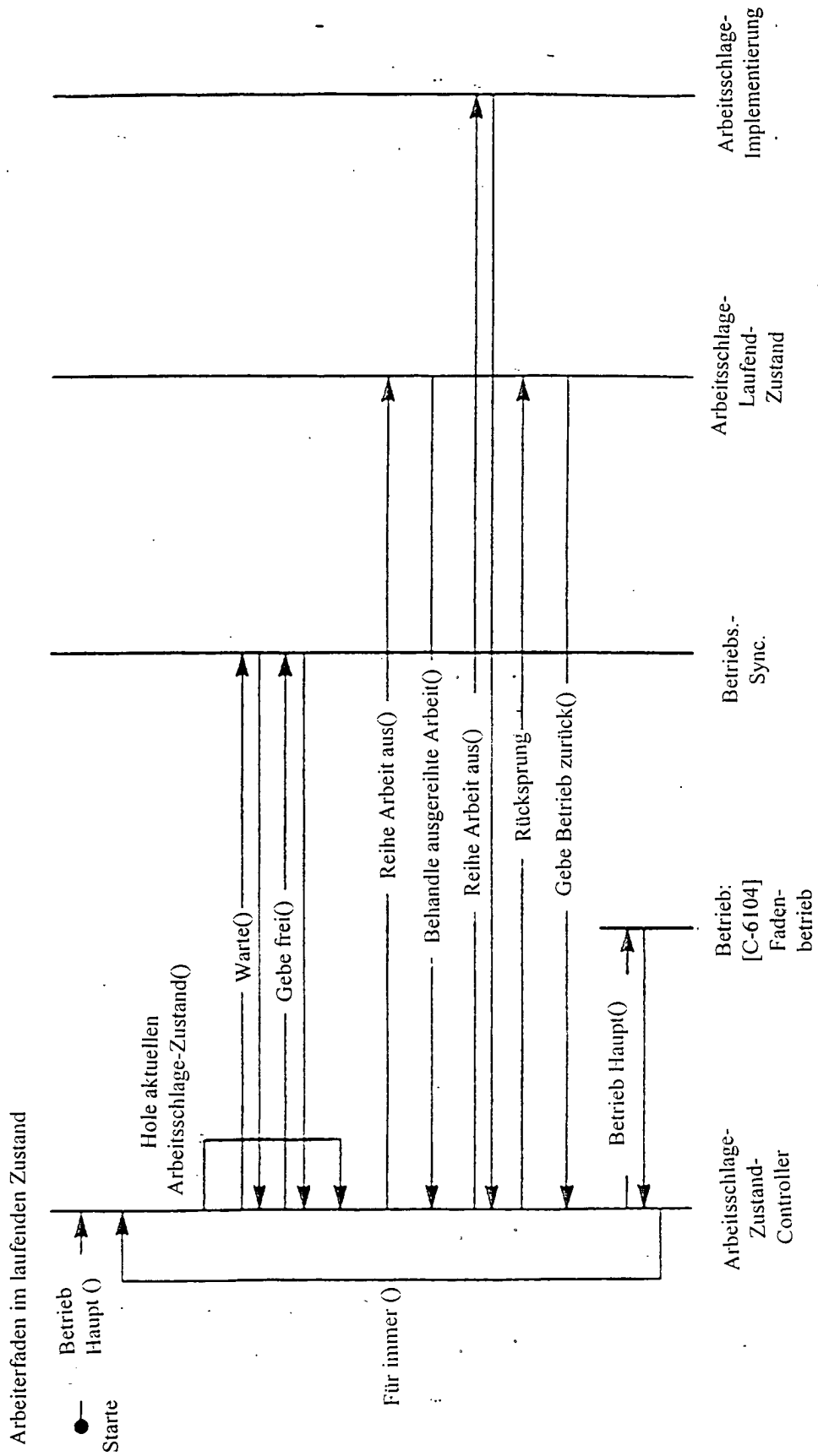


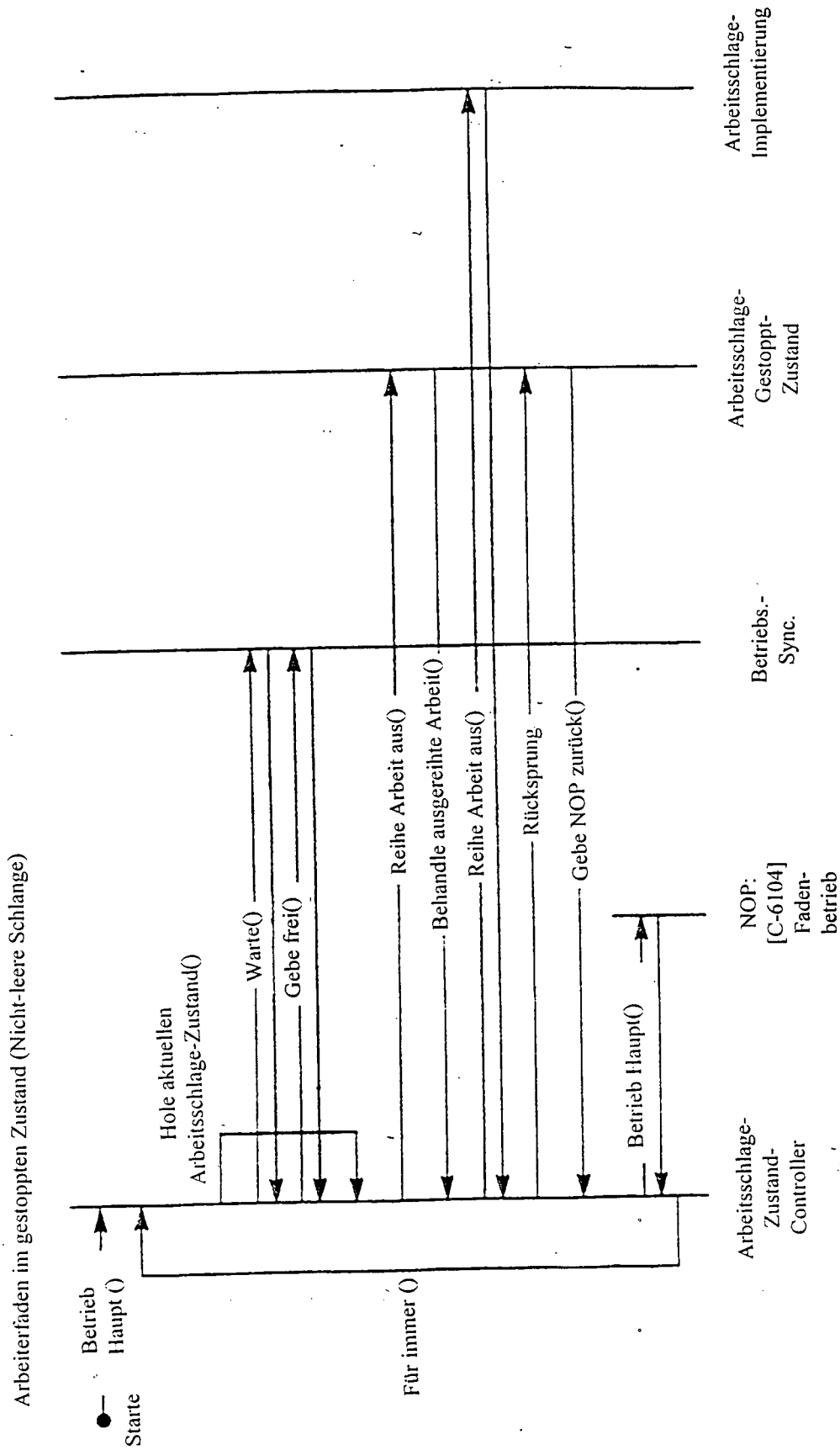


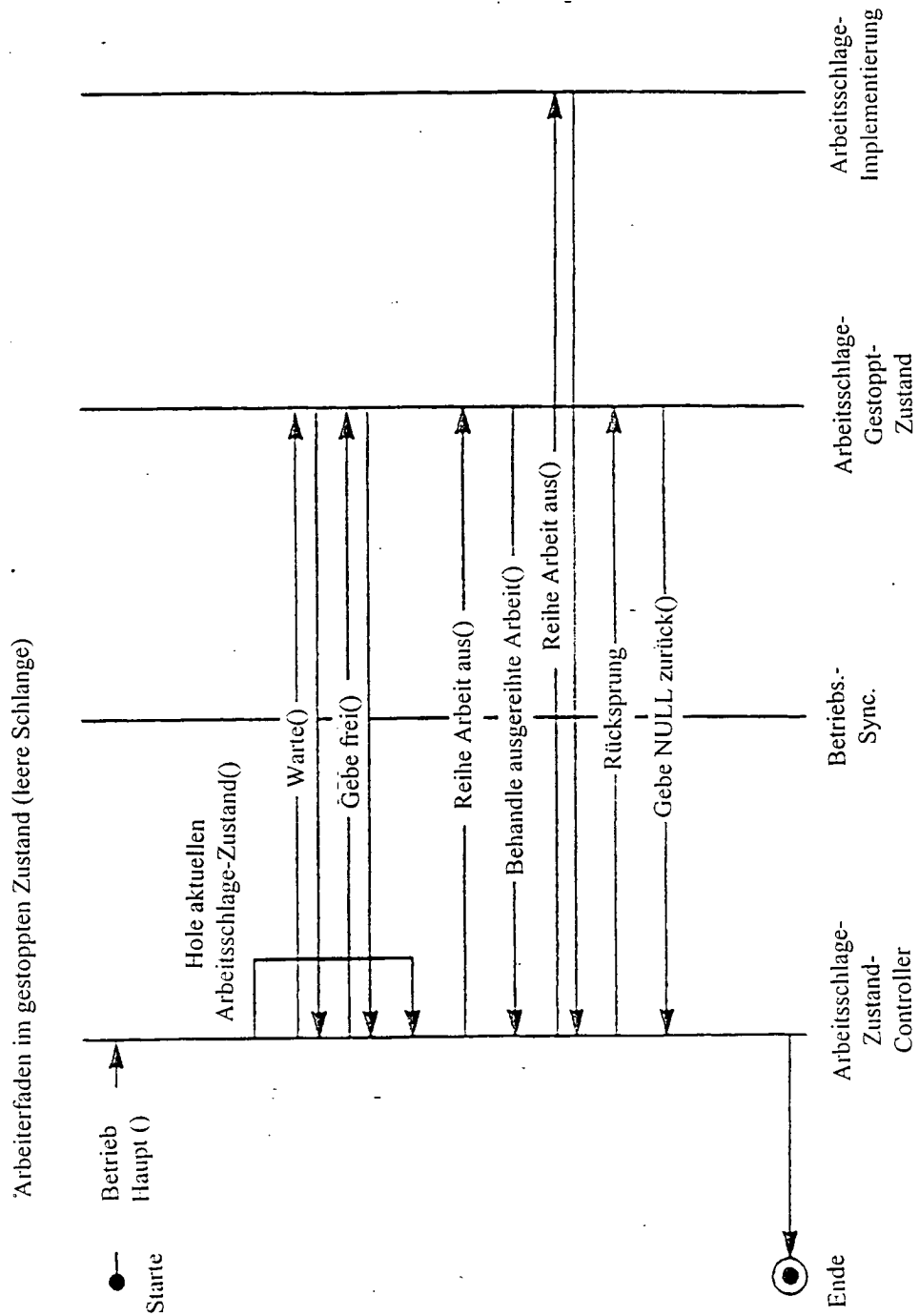




Figur 10

**Figur 11**

**Figur 12**

**Figur 13**