



MCT Task

Analyse

Erstellt durch:

- Markus Bättig, Schulstrasse 13, 6037 Root
- Jürg Zraggen, Ottigenbühlring 11, 6030 Ebikon

Heutiges Batchsystem

Beschreibung

MRTG wird mittels dem Task Scheduler auf dem Server gestartet. In einem Parameter wird das gewünschte Konfigurationsfile MRTG übergeben.

Bsp: c:\perl\bin\perl.exe MRTG -cfg=c:\temp\test.cfg

Die Firma X arbeitet heute mittels Batchdateien, um den Windows Task Scheduler aktuell zu halten. In diesen Dateien befindet sich meistens auf der ersten Zeile der Befehl "at.exe /delete". Dieser Befehl bewirkt das Löschen ALLER Task Einträge. Danach werden die gewünschten wieder neu erstellt.

So eine Datei kann folgendermassen aussehen:

```
@echo off  
at /delete  
at.....
```

Diese Methode der Taskkontrolle kann schon bei sehr kleinen Anpassungen recht aufwendig und umständlich werden. Die Batchdatei muss manuell editiert und syntaktisch korrekt gespeichert werden. Danach muss sie mit den nötigen Rechten auf dem entsprechenden Server ausgeführt werden. Frühere Tasks werden meist gelöscht und neu erstellt. Auch birgt dies ein grosses Risiko, mal einen Task zu erwischen, welcher nicht zu MRTG gehört. Genau dort möchten wir den User bei dieser Arbeit unterstützen. Um die Kontrolle über diese MRTG-Tasks abzudecken, soll ein Programm MCT Task erstellt werden, welches über die Webseite ansprechbar sein wird. Das Programm soll mit C++ und Konsolenoutput realisiert werden, da hier einen grosser Teil an Windows-API-Funktionen verwendet wird.

MCT Task

Beschreibung

MCT Task wird der obigen beschriebenen Problematik vorzubeugen, indem MCT über PHP direkt auf die Windows Tasks Einfluss nehmen kann. Das Programm managed den Windows Task Scheduler für MCT in kontrollierter Art und Weise. Das Programm soll bestehende Tasks löschen od. neue hinzufügen können.

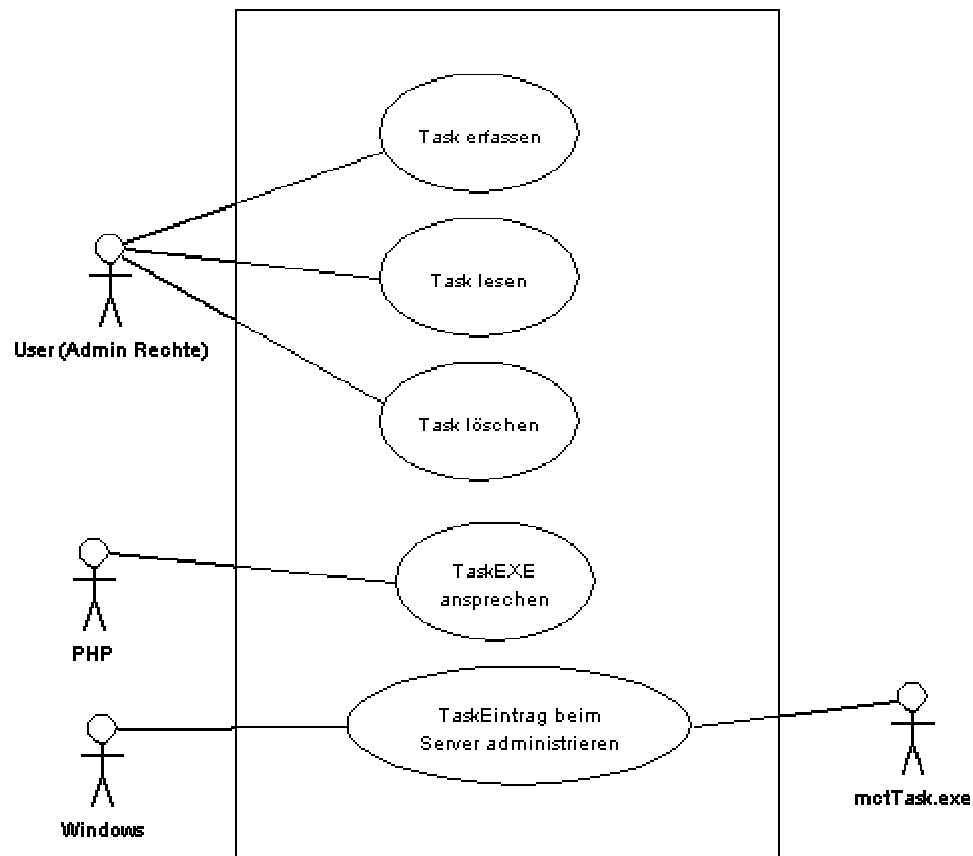
Das Programm erstellt jedem Task einen Präfix "mct_", um ihn als MCT-Task zu identifizieren. Diesen Präfix kann im MCT.ini geändert werden.

Die Software hat keine direkte Datenbankanbindung und muss somit alle benötigten Informationen über die Parameterliste und über ein Definitionsfile bekommen. Da recht viele Parameter mit Pfadangaben übergeben werden müssten, haben wir uns für ein Definitionsfile entschieden. Dies ist übersichtlicher und skalierbarer. Nur der Ort der gespeicherten Definitionsdatei wird der EXE mittels Parameter übergeben.

Die in dem Definitionsfile verfügbaren Parameter sind genau bestimmt und haben klare Grenzwerte definiert, die nicht über- resp. unterschritten werden dürfen. Gibt man somit im Definitionsfile unmögliche oder fehlerhafte Werte ein, die gemäss Grenzwert nicht sein dürfen, kann das Programm ungewöhnlich und fehlerhaft reagieren. Da jedoch diese MCT Task Software nur von der MCT Webseite gebraucht und angesprochen wird, werden diese Fälle mit grosser Wahrscheinlichkeit nicht eintreten, da die Webseite ja weiss, wie sie das Programm anzusprechen hat und sich an die Vorgaben halten wird. Schlussendlich liefert das Programm einen Output zurück, ob und wie alles geklappt hat. Dieser Output wird vom PHP von der Konsole gecaptured im und dem User als Debug-Informationen auf dem Web dargestellt.

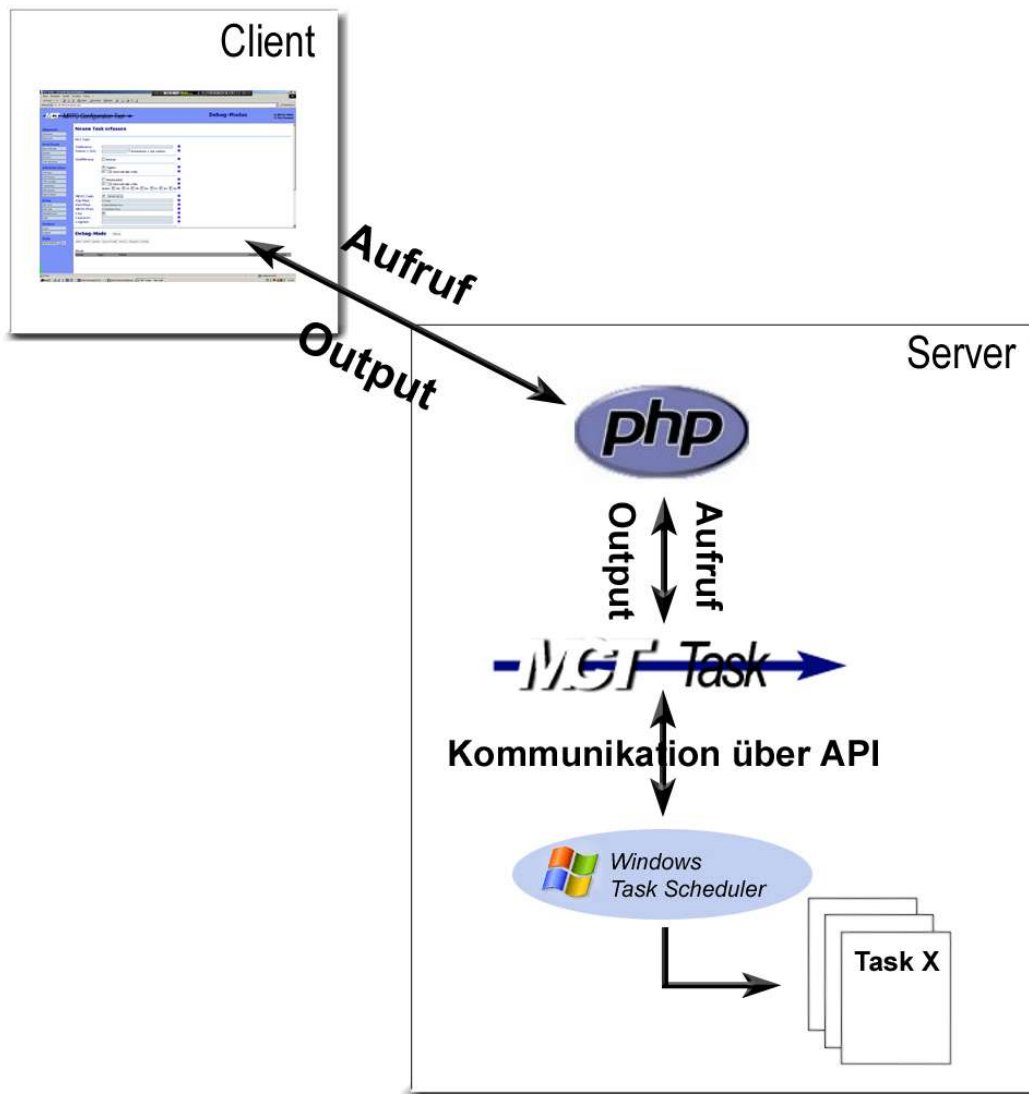
Use Case Diagramm

In diesem Use Case Diagramm wird beschrieben, in welchen Bereichen welche Aktoren Zugriff haben resp. Tätig sind.



Ablaufdiagramm

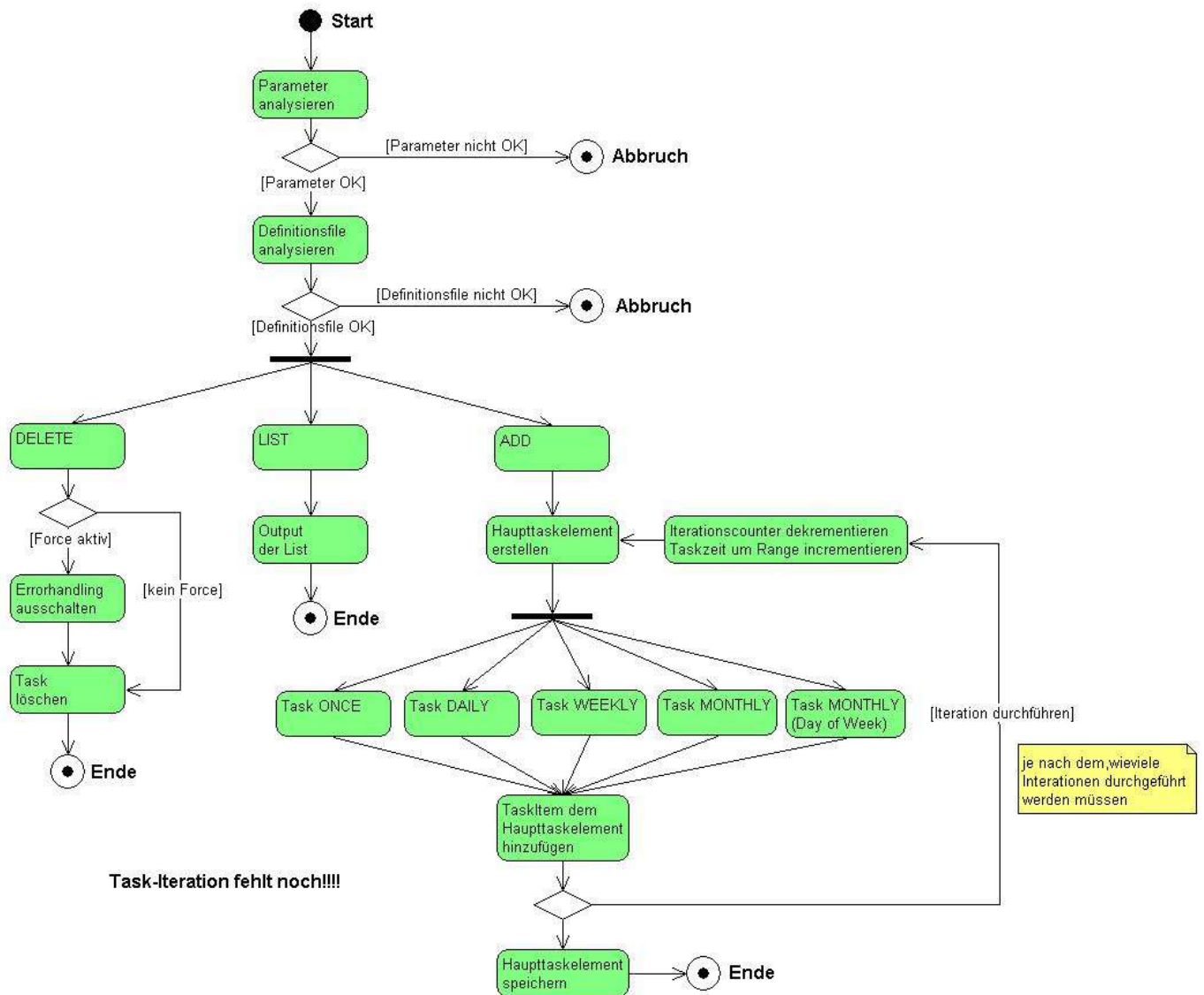
Auf der MCT Webseite wird ein Definitionsfile geschrieben und mittels Funktionsaufruf PHP angefordert die MCT_Task.exe mit den nötigen Parameter zu starten. PHP nimmt diesen Befehl entgegen und ruft nun MCT_Task.exe auf. MCT_Task.exe startet, liest das Definitionsfile und prüft die bekommenen Parameter auf Richtigkeit. Danach agiert es je nach Befehl und administriert alle MCT-Tasks im Windows Task Scheduler über die Windows API. Diese API liefert der MCT_Task.exe zurück, ob die Aktion erfolgreich war oder nicht. Nachdem die MCT_Task.exe ihre Arbeit erledigt hat und terminiert, captured PHP den Konsolenoutput der MCT_Task.exe und übermittelt diesen der MCT Webapplikation. Diese stellt den Output nun dem Debug-Objekt zur Verfügung und gibt dem Benutzer je nach dem eine Erfolgreich- oder Error-Meldung aus.



Aktivitätsdiagramm der MCT_Task.exe

Dieses Aktivitätendiagramm zeigt den internen Ablauf der MCT_Task.exe auf.

Aktivitätsdiagramm TaskEXE



Programm Parameter

-def=	Definiert den Dateipfad, wo sich die Definitionsdatei für MCT Task.exe befindet
-?	Fordert die MCT_Task.exe auf, die Parameterhilfe anzuzeigen. Danach wird das Programm terminiert ohne eine Aktion zu tätigen.

Definitionsdatei

Das MCT Definitionsfile wird als INI dargestellt. Je nach Aktion kann das INI unterschiedlich aussehen. Die mit [optional] gekennzeichneten Parameter können bei Bedarf auch weggelassen werden. Um dies zu verdeutlichen, haben wir von den 3 Haupttätigkeiten auflisten, löschen und hinzufügen je 1 Beispiele aufgeführt.

Tasks auflisten:

```
[MCT_TASK]

[HEADER]
action=list
taskname=mct*           ' Korrekter Name des Tasks oder mit Stern für folgende
delafterexec=true       ' [optional] Definitionsfile nach dem Ausführen löschen
```

Task löschen:

```
[MCT_TASK]

[HEADER]
action=del
taskname= mct*          ' Korrekter Name des Tasks oder mit Stern für folgende
delafterexec= true      ' [optional] Definitionsfile nach dem Ausführen löschen

[ACTION_DEL]
force=true              ' [optional] Zeigt Errors beim Löschen an, stop das Programm
                        ' jedoch nicht!
```

Task hinzufügen:

```
[MCT_TASK]

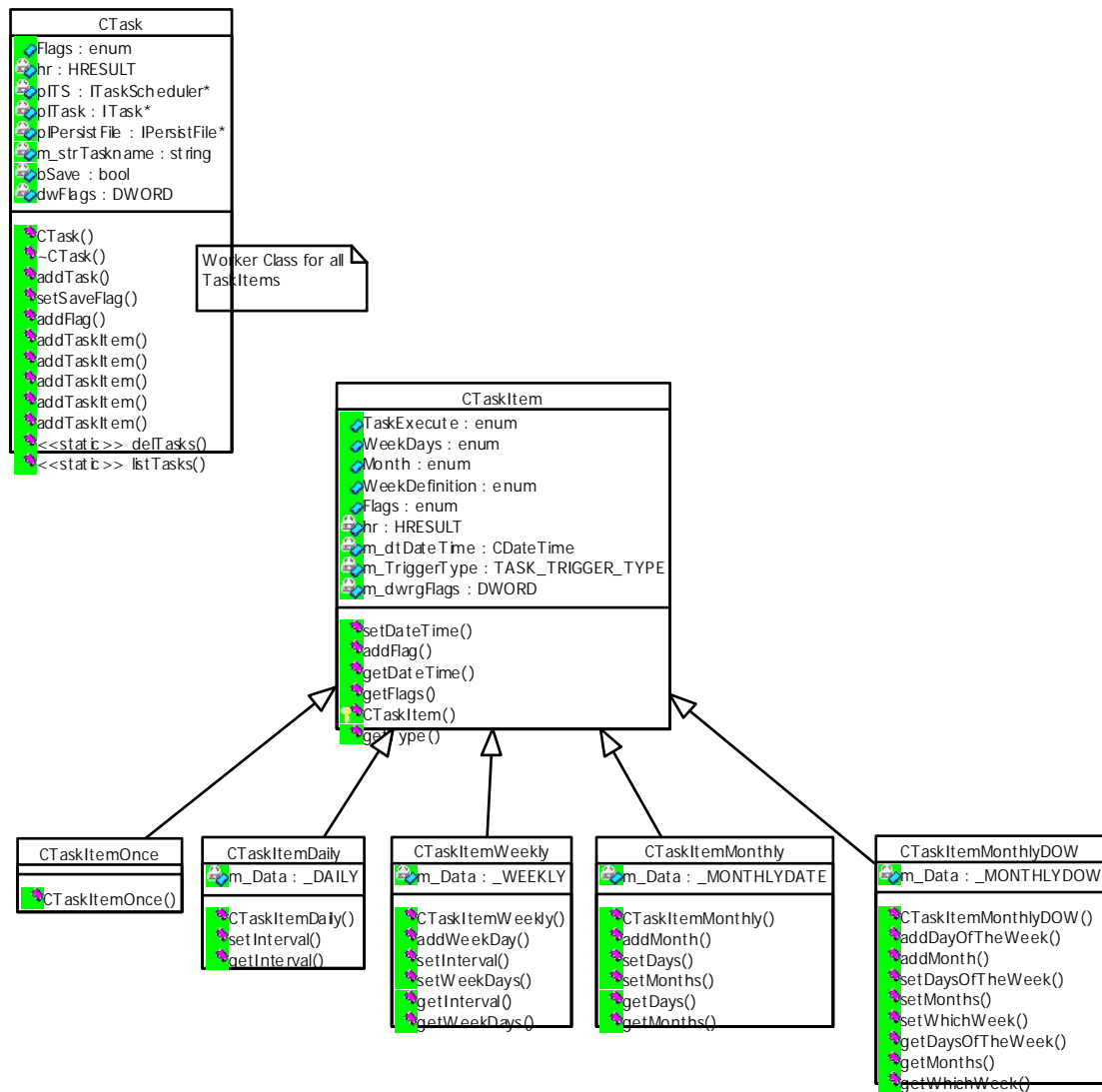
[HEADER]
action=add
taskname=mct_TaskX      ' Name des hinzuzufügenden Tasks
delafterexec= true      ' [optional] Definitionsfile nach dem Ausführen löschen

[ACTION_ADD]
type= daily              ' Mögliche Typen: once / daily / weekly / monthly / monthlyDOW
executable=c: \notepad.exe ' Pfad des Programmes, welches ausgeführt werden soll.
date=2003-12-12         ' Startdatum des Taskes
time=11:00               ' Startzeit des Taskes
account=jrg              ' Windows-Account mit den nötigen Berechtigungen
pwd=zg                   ' Passwort des Windows-Accountes
```

creator=MCT	' [optional] Task-Ersteller
comment=no comment	' [optional] Task Kommentar
parameter=	' [optional] Parameter der auszuführenden Exe
workpath=	' [optional] Arbeitsverzeichnis der auszuführenden Exe
unique=false	' [optional] Definiert, ob ein einzigartiger Taskname generiert werden soll
timeiteration=3	' [optional] Anzahl Iterationen, die durchgeführt werden sollen. ' Die letzte Zeit wird bei jeder Iteration um den timerange erhöht. ' Somit erhält man die Möglichkeit, dass ein Task um ' verschiedene Zeiten gestartet wird.
timerange=10	' [optional] Zeit in Minuten, die bei jeder Iteration addiert werden soll. Bsp: 10 (Task um 17:00, 17:10, 17:20, 17:30)
interval=2	' [daily, weekly] Regelmässigkeit (Serienmuster), die durchgeführt werden sollen. Bsp: 2 (jeden 2ten Tag)
days=	' [monthly, monthlyDOW] Kommaseparierte Liste, an welchen Tagen der Task ausgeführt werden soll. Bei monthlyDOW ist nur ein Wert erlaubt ' (1=Mo, 2=Di, 3=Mi, ..., 7=So) Bsp: 2,5,7
months=	' [monthly, monthlyDOW] Kommaseparierte Liste, an welchen Monaten der Task ausgeführt werden soll ' (1=Jan, 2=Feb, ..., 12=Dez) Bsp: 2,5,7
whichweek=	' [monthlyDOW] Definition, welche Woche gemeint ist. ' first, second, third, fourth, last

Klassendiagramm

Der detaillierte Klassenbeschrieb ist dem Anhang zu entnehmen!



Entwicklung MCT Task

Quellen:

Diverse Code-Stücke konnten von der Microsoft Homepage übernommen werden. Sie mussten nur noch zu einer funktionierenden Klasse abgeändert und zusätzliche Logik angefügt werden.

http://msdn.microsoft.com/library/en-us/taskschd/taskschd/task_scheduler_start_page.asp

Vorhandene Hilfsklassen:

Die folgenden Klassen mussten nicht mehr erstellt werden. Sie stammen von früheren Projekten der IHA-GfK und dürfen von uns verwendet werden. Sie gehören somit nicht zum Leistungsumfang dieses Projektes.

- CCmdLine
- SParameterPair
- CDateTime
- CException
- CEx_FileNotOpen
- CProtocol
- CStatistic
- CStatisticIO
- CStatisticDB
- CTools
- CConverter
- CcharParser

Inhaltsverzeichnis

Analyse	1
Heutiges Batchsystem.....	2
Beschreibung.....	2
MCT Task.....	3
Beschreibung.....	3
Use Case Diagramm	4
Ablaufdiagramm	5
Aktivitätsdiagramm der MCT_Task.exe	6
Programm Parameter.....	7
Definitionsdatei.....	7
Tasks auflisten:	7
Task löschen:	7
Task hinzufügen:	7
Klassendiagramm	9
Entwicklung MCT Task	10
Quellen:.....	10
Vorhandene Hilfsklassen:.....	10
Inhaltsverzeichnis.....	11