



User's Guide
RSDS

SIMONA report number ra/30



User's Guide RSDS

RSDS is a program to read
contents from an SDS-file

Version : 1.5, Juli 2008
Maintenance : zie www.helpdeskwater.nl/waqua
Copyright : Rijkswaterstaat

DOCUMENTGESCHIEDENIS

Versie	Datum	simona/ra	Omschrijving
1.0	30-05- 1997	17/BCV	Eerste uitgave, vrijgegeven door SIMTECH
1.1	01-09-1997	30/BCV	Tweede uitgave

VOORWOORD

Met behulp van het programma RSDS (Read Simona Data Storage file) kunnen alle gegevens, die zijn opgeslagen in een SDS file, worden opgevraagd en worden weggeschreven in een uitvoer formaat dat door de gebruiker zelf kan worden bepaald.

INHOUDSOPGAVE

1	INLEIDING	1
1.1	RSDS in het kort	1
1.2	Doel van de handleiding	1
1.3	Opzet van de handleiding	2
1.4	Notaties en conventies.....	2
2	GEBRUIK VAN RSDS.....	7
2.1	Starten van RSDS.....	7
2.1.1	<i>Op een HP-9000 machine</i>	<i>7</i>
2.1.2	<i>Op een PC.....</i>	<i>10</i>
2.2	Invoer en uitvoer van RSDS.....	11
2.3	RSDS opdrachten	13
2.3.1	<i>Overzicht.....</i>	<i>13</i>
2.3.2	<i>Conventies.....</i>	<i>15</i>
2.3.3	<i>Beschrijving.....</i>	<i>16</i>
2.3.4	<i>Foutmeldingen.....</i>	<i>34</i>
2.4	RSDS in batch.....	37
2.4.1	<i>Met behulp van een batch file.....</i>	<i>37</i>
2.4.2	<i>Met behulp van batch commando's</i>	<i>38</i>
3	TECHNISCHE ACHTERGROND RSDS	39
3.1	De SDS documentatie file	39
3.2	Het directory systeem DIRSYS	40
4	GEWENSTE UITBREIDINGEN EN BEPERKINGEN	41
4.1	Gewenste uitbreidingen.....	41
4.2	Beperkingen van RSDS	41
5	REFERENTIES	49
6	VOORBEELDEN INVOER RSDS	51
6.1	Voorbeeld: dimen.inp.....	52
6.2	Voorbeeld: formats.inp	55
6.3	Voorbeeld: binair.inp.....	57
6.4	Voorbeeld: bndout.inp	58
6.5	Voorbeeld: cco.inp.....	59
6.6	Voorbeeld: lagen.inp	61
6.7	Voorbeeld: stations.inp.....	63
6.8	Voorbeeld: tijden.inp.....	64
6.9	Run-script: rlds.gain	65
6.10	Run-script: rlds.tijd	67
7	VOORBEELDEN SDS DOCUMENTATIE FILES.....	69
7.1	SDS documentatie file voor WAQUA.....	69
7.2	SDS documentatie file voor WAQAD	83
7.3	SDS documentatie file voor KALMAN	87
7.4	SDS documentatie file voor meetwaarden.....	88
7.5	SDS documentatie file voor wind.....	90

1 INLEIDING

In dit hoofdstuk wordt eerst een korte toelichting gegeven op het programma RSDS (paragraaf 1.1). Vervolgens worden het doel van de handleiding (paragraaf 1.2) en de opzet van de handleiding (paragraaf 1.3) aangegeven. Als laatste wordt een overzicht gegeven van de in deze handleiding gebruikte notaties en conventies (paragraaf 1.4).

1.1 RSDS in het kort

Bij Rijkswaterstaat zijn meerdere applicaties volgens de SIMONA norm geprogrammeerd, zoals WAQUA. Veelal wordt hierin de uitvoer weggeschreven naar een SDS file (Simona Data Storage file). Dit is een binaire file waaruit de opgeslagen gegevens gelezen kunnen worden met behulp van standaard naverwerkingsprogrammatuur. Dit levert echter de beperking dat slechts door de programmatuur geselecteerde gegevens kunnen worden opgevraagd ten behoeve van door die programmatuur vastgelegde naverwerking.

Met behulp van het programma RSDS (Read SDS file) kunnen alle gegevens, die zijn opgeslagen in een SDS file, worden opgevraagd. Deze gegevens kunnen met behulp van RSDS zowel in ascii formaat als binair worden weggeschreven. Het uitvoer formaat, waarmee de gegevens weg worden geschreven, kan door de gebruiker zelf worden bepaald. Interactie met de gebruiker kan zowel on-line als in batch zijn. Zo is eenvoudig een interface te realiseren tussen een SDS file en andere programmatuur, zoals Matlab, Lotus, etc.

1.2 Doel van de handleiding

Deze gebruikershandleiding is bedoeld voor diegenen die:

- RSDS willen installeren op een HP-9000 machine of op een PC,
- RSDS willen gebruiken op een HP-9000 machine of op een PC.

Naast een beschrijving, wordt het gebruik van RSDS duidelijk gemaakt door middel van een aantal voorbeelden die zijn opgenomen in deze handleiding. De voorbeelden zijn ook beschikbaar op diskette, zodat de gebruiker deze direct kan toepassen.

Verder wordt het installeren en gebruiken van RSDS overzichtelijk gemaakt door middel van samenvattingen die zijn opgenomen aan het eind van een aantal paragrafen. Deze samenvattingen kunnen als naslag dienen voor de gevorderde RSDS gebruiker.

1.3 Opzet van de handleiding

De gebruikershandleiding heeft de volgende hoofdstuk opbouw:

- hoofdstuk 1 korte beschrijving van RSDS, het doel en de opzet van de handleiding en de gebruikte notaties en conventies
- hoofdstuk 2 gebruik van RSDS: starten met RSDS, invoer en uitvoer van RSDS, het geven van opdrachten, RSDS in batch uitvoeren
- hoofdstuk 3 de technische achtergrond van RSD: toelichting op de SDS documentatie file en op DIRSYS
- hoofdstuk 4 overzicht van beperkingen en gewenste uitbreidingen van RSDS
- hoofdstuk 5 referenties
- hoofdstuk 6,7 voorbeelden van invoer files en run-scripts die direct toegepast kunnen worden

1.4 Notaties en conventies

In deze handleiding worden de volgende notaties gebruikt:

keyword	Een keyword in vet gedrukte kleine letters in de kantlijn is het "sleutelwoord" waar het in de betreffende paragraaf om gaat.
OPDRACHT	Vet gedrukte kleine hoofdletters in de kantlijn betreffen de opdrachten die gegeven kunnen worden in het programma RSDS. Opdrachten kunnen gevolgd worden door invoer bestaande uit integers, reals of characters. Conventies voor de invoer van opdrachten worden beschreven in subparagraaf 2.3.2.
Tekst	Als een bepaalde tekst moet worden opgenomen in een bestand, dan wordt deze tekst aangegeven in een ander lettertype. Deze tekst moet dan letterlijk, met onderscheid naar hoofd- en kleine letters in het betreffende bestand worden opgenomen.
uitvoer	De RSDS uitvoer, opgenomen in dit rapport als voorbeeld, wordt op dezelfde wijze weergegeven: in een ander lettertype.
"bestand"	Voorbeelden van machines, directories en bestanden worden aangegeven met dubbele quotes. Bijvoorbeeld: het bestand "rsds.pl" op de directory "/users/applic/waqad/bin" op het rekencluster "compcl".

<omschrijving>

Op deze manier wordt aangegeven dat de gebruiker een bepaalde tekst moet opnemen of intikken waarbij de omschrijving door de gebruiker zelf wordt ingevuld. Als bijvoorbeeld wordt aangegeven dat de volgende tekst moet worden opgenomen:

SIMONADIR=<SIMONA directory>

dan moet de gebruiker achter het "=" teken de SIMONA directory opgeven, bijvoorbeeld:

SIMONADIR=/users/applic/simona

Opmerking In de tekst kunnen zowel Engelse termen als Nederlandse termen voor hetzelfde begrip voorkomen, afhankelijk van de terminologie die bij Rijkswaterstaat gehanteerd wordt. Zo worden bijvoorbeeld de termen "file" en "bestand" door elkaar gebruikt.

2 GEBRUIK VAN RSDS

In dit hoofdstuk wordt beschreven hoe RSDS gestart wordt (paragraaf 2.1), hoe de invoer en uitvoer eruit zien (paragraaf 2.2), welke opdrachten gegeven kunnen worden (paragraaf 2.3) en hoe RSDS in batch uitgevoerd kan worden (paragraaf 2.4).

2.1 Starten van RSDS

In de volgende subparagrafen wordt beschreven hoe RSDS wordt gestart op een HP-9000 machine en op een PC.

2.1.1 Op een HP-9000 machine

starten rds

Het programma RSDS wordt gestart door het uitvoeren van de procedure:

rsds.pl

Na het opgeven van dit commando verschijnen de volgende kopregels:

```
*****
runprocedure R S D S
Read information from the SDS file
*****
```

run id

Vervolgens vraagt RSDS om de run identificatie, dit is de "opc" van de SDS file:

```
Give run identification?
```

De gebruiker voert nu de run identificatie in gevolgd door een return. Als er geen run identificatie is ingevoerd, dan volgt een melding en wordt opnieuw om de run identificatie gevraagd:

```
*** Error: no run identification given!
Give run identification?
```

Als een verkeerde run identificatie is ingevoerd, dan volgt een melding en wordt opnieuw om de run identificatie gevraagd:

```
*** Error: SDS-<opc> does not exist!
Give run identification?
```

De SDS file waarvan de run identificatie is opgegeven moet zich wel op de huidige directory bevinden, anders geeft RSDS ook de bovenstaande foutmelding.

invoer

Na het opgeven van de run identificatie wordt gevraagd om de naam van de invoer file:

```
Give input filename (<RETURN> = HAND)?
```

De gebruiker voert de naam van de invoer file in gevolgd door een return, of alleen een return indien de invoer handmatig wordt opgegeven. De naam van de invoer file kan ook vooraf worden gegaan door een (volledig) pad indien de invoer file zich niet in de huidige directory bevindt. Als een verkeerde invoer file is opgegeven, dan volgt een melding en wordt opnieuw om de naam van de invoer file gevraagd:

```
*** Error: <invoerfile> does noet exist!  
Give input filename (<RETURN> = HAND)?
```

uitvoer

Na het opgeven van de invoerfile wordt om de naam van de uitvoerfile gevraagd:

```
Give output filename (<RETURN> = STDOUT)?
```

De gebruiker voert de naam van de uitvoer file in gevolgd door een return, of alleen een return indien standaard uitvoer, dit is uitvoer naar scherm, wordt weggeschreven. De naam van de uitvoer file kan ook vooraf worden gegaan door een (volledig) pad indien de uitvoer file niet naar de huidige directory geschreven moet worden.

background

Als bij het opgeven van de invoer de naam van een invoer file is ingetikt, dan wordt nu gevraagd of RSDS in de achtergrond gedraaid moet worden:

```
RSDS program run in the background (y/n)?
```

Als RSDS in de achtergrond moet worden uitgevoerd, dan moet achter deze vraag een "y" worden ingetikt. Als RSDS in de voorgrond moet worden uitgevoerd, dan kan een "n" worden ingetikt of kan een return worden gegeven. Als iets anders dan "n", een return of een "y" wordt opgegeven, dan wordt RSDS ook in de voorgrond uitgevoerd. Als de invoer handmatig wordt opgegeven, dan kan RSDS niet in de achtergrond worden uitgevoerd en wordt deze vraag niet gesteld.

Na het eventueel opgeven of RSDS in de achtergrond moet worden gedraaid, opent RSDS de SDS file en het enig beschikbare experiment. Dan verschijnen de naam van de opgegeven SDS file en het opgegeven experiment:

```
-----
SDS file           : SDS-<opc>
Experiment        : <exp>
-----
```

Als de naam van een invoer file is opgegeven, dan worden de opdrachten in die invoer file door RSDS uitgevoerd en stopt het programma zodra het eind van de invoer file bereikt is. Als handmatige invoer wordt opgegeven, dan volgt nu de RSDS prompt:

```
SI>
```

Achter deze prompt kunnen RSDS opdrachten worden gegeven. Meer informatie over invoer, uitvoer en het geven van opdrachten bevindt zich in de volgende paragrafen.

Samenvatting:

- **rds.pl**
- opgeven run identificatie
- opgeven invoer
- opgeven uitvoer
- (opgeven RSDS in achtergrond/voorgrond)

Overzicht scherm:

```
*****  
  
runprocedure R S D S  
Read information from the SDS file  
  
*****  
  
Give run identification?  
Give input filename (<RETURN> = HAND)?  
Give output filename (<RETURN> = STDOUT)?  
(RSDS program run in the background (y/n)?)  
-----  
SDS file : SDS-<opc>  
Experiment : <exp>  
-----  
  
SI>
```

2.1.2 Op een PC

Het starten van RSDS op de PC gaat in principe op dezelfde manier als starten van RSDS op een HP-9000 machine. Deze sub-paragraaf geeft daarom alleen de verschillen aan ten opzichte van het starten van RSDS op een HP-9000 machine.

Omdat op dit moment RSDS nog niet op een PC is geïnstalleerd, volgt meer informatie in een nieuwe versie van deze gebruikershandleiding.

2.2 Invoer en uitvoer van RSDS

Er zijn twee mogelijkheden voor het geven van invoer aan RSDS:

- via een invoer file
- handmatig

invoer

Als de invoer voor RSDS wordt opgegeven door middel van een invoer file, dan moet bij het starten van RSDS de naam worden opgegeven van deze invoer file eventueel voorafgegaan door een (volledig) pad. In de invoer file bevinden zich opdrachten die door RSDS kunnen worden uitgevoerd (zie paragraaf 2.3). Na het starten van RSDS worden de opdrachten in de invoer file uitgevoerd en stopt RSDS zodra het einde van de file bereikt is.

Als de invoer voor RSDS handmatig wordt opgegeven, dan moet bij het starten van RSDS een return worden gegeven na de vraag om de invoer file. In dit geval toont RSDS een prompt en wacht tot een opdracht door de gebruiker wordt ingevoerd. Het programma stopt na opgeven van de opdracht **EXIT**, zie paragraaf 2.3.

uitvoer

De uitvoer kan door RSDS op twee manieren worden weggeschreven:

- naar standaard uitvoer (dit is naar scherm, of naar een uitvoer file, indien gebruik wordt gemaakt van "redirection")
- naar uitvoer file

Als voor standaard uitvoer is gekozen, dan moet bij het starten van RSDS een return worden gegeven na de vraag om de uitvoer file. In dit geval wordt de uitvoer van RSDS naar het scherm geschreven. Het is echter toch mogelijk om bepaalde gegevens, bijvoorbeeld de waarden van variabelen in de SDS file, naar één of meerdere uitvoer files te schrijven. Dit kan met behulp van de opdracht **OUTFILE**, zie paragraaf 2.3.

Als na de vraag om de uitvoer file de naam van een uitvoer file wordt opgegeven eventueel voorafgegaan door een (volledig) pad, dan wordt de standaard uitvoer niet naar scherm geschreven maar naar de opgegeven uitvoer file. Het is dan echter toch mogelijk om bepaalde gegevens naar één of meerdere, andere, uitvoer files te schrijven met behulp van de opdracht **OUTFILE**, zie paragraaf 2.3.

In totaal zijn er dus vier mogelijkheden voor het opgeven van invoer en uitvoer van RSDS:

Handmatige invoer, uitvoer naar scherm:

Deze optie wordt voornamelijk gebruikt bij testen. Om bijvoorbeeld even snel de waarde van een bepaalde variabele op de SDS file na te gaan. Deze optie wordt niet aangeraden indien (routinematig) veel uitvoer wordt gegenereerd, aangezien het handmatig invoeren van de benodigde opdrachten tijdrovend kan zijn. Deze optie kan in batch worden gestart, RSDS wordt echter wel on-line uitgevoerd, zie paragraaf 2.4.

Invoer file, uitvoer naar scherm:

Deze optie wordt het meest gebruikt. Deze optie is namelijk geschikt voor het automatisch genereren van grote hoeveelheden uitvoer, waarbij de

meeste gegevens toch naar één of meerdere uitvoer files kunnen worden geschreven. De overige gegevens worden als standaard uitvoer naar het scherm geschreven.

Invoer file, uitvoer file:

Deze optie komt grotendeels overeen met de vorige. De standaard uitvoer wordt echter niet naar scherm geschreven maar naar de uitvoer file die is opgegeven bij het starten van RSDS.

Handmatig invoer, uitvoer file:

Indien deze optie wordt gebruikt, dan wordt de standaard uitvoer naar file geschreven. Dit betekent echter dat de RSDS prompt, die als standaard uitvoer wordt gezien, ook naar de uitvoer file wordt geschreven. Deze optie is niet makkelijk in het gebruik en wordt daardoor niet aangeraden.

2.3 RSDS opdrachten

In de volgende sub-paragrafen komen aan de orde: een overzicht van de opdrachten die gegeven kunnen worden in RSDS, de conventies van de opdrachten, een beschrijving van de opdrachten, en een beschrijving van de foutmeldingen.

2.3.1 Overzicht

Hieronder volgt een overzicht van de opdrachten die gegeven kunnen worden in RSDS. De opdrachten zijn niet alfabetisch gerangschikt, maar zijn weergegeven voor de volgende categorieën:

- variabele
- uitvoer algemeen
- uitvoer ascii
- uitvoer matlab
- opties

variabele

Instellen van variabele:

NAME	Het opgeven van de naam van de variabele
LGRID	Aangeven of de variabele een lgrid index heeft
START	Opgeven van het begin adres van de variabele
LENGTH	Opgeven van de lengte van de variabele
DIMEN	Opgeven van de dimensie van de variabele (2-dimensionaal)

algemene uitvoer

Instellen van uitvoer (algemeen):

MNMN	De begin- en eind-coördinaten van een veld voor het wegschrijven van de waarden van een variabele
TIME	Tijdstip(pen) waarop de waarden van een variabele gelezen en weggeschreven worden
TYPE	Opgeven van een type (integer/real/character) voor het wegschrijven van de waarden van een variabele in de uitvoer
REPEAT	Herhaal factor voor het wegschrijven van de waarden van een variabele
DUMVAL	Waarde waarmee dummies in lgrid-structuur worden vervangen
GO	Het lezen en wegschrijven van de waarden van een variabele

ascii uitvoer

Instellen van ascii uitvoer:

BOX	Het aanzetten van de box-mode: de uitvoer wordt weggeschreven in box-formaat
NOBOX	Het uitzetten van de box-mode
MATLAB	Het aanzetten van de matlab-mode: de uitvoer wordt weggeschreven in matlab-formaat, d.w.z. in een script dat in MATLAB kan worden ingelezen en uitgevoerd.
NOMATLAB	Het uitzetten van de matlab-mode
PRINT	Het printen van een opgegeven tekst naar uitvoer file
FORMAT	Het opgeven van een format voor het wegschrijven van de waarden van variabelen in de uitvoer
PRITIM	Het opgeven van een format voor het wegschrijven van de tijden in

OUTFILE	de uitvoer Het opgeven van de naam van een ascii uitvoer file
binaire matlab uitvoer	Instellen van binaire matlab uitvoer:
MATFILE	Het opgeven van de naam van de matlab uitvoer file
MATNAME	Het opgeven van de matlab naam van een variabele
MATTIME	Het wegschrijven van tijdreeksen in de matlab uitvoer file
opties	Instellen opties:
SHOW	Het laten zien van de actuele instellingen
DEFAULT	Het initialiseren van een aantal instellingen met behulp van default waarden
TREE	Het laten zien van de directories en variabelen in de SDS file
NLEVEL	Het aantal levels dat getoont wordt m.b.v. de opdracht TREE
ECHO	Het schrijven van de invoer op de command line naar standaard output
REOPEN	Het opnieuw openen van de SDS file
EXIT	Het verlaten van RSDS

2.3.2 Conventies

Een aantal RSDS opdrachten moet worden gevolgd door bepaalde invoer zoals bijvoorbeeld een string, integer(s) of real(s). De syntax voor het geven van opdrachten en het meegegeven van invoer aan de opdrachten is als volgt (zie ook "Programmer's guide SIMONA", [1]):

- Een opdracht bestaat uit een aantal alfanumerieke karakters. De opdrachten kunnen zowel in hoofdletters als in kleine letters worden opgegeven. Slechts een aantal karakters in de opdrachten zijn significant, deze worden bij de beschrijving (paragraaf 2.3.3) onderlijnd. De opdracht LGRID kan bijvoorbeeld met behulp van de volgende (zelfde) commando's gegeven worden:

```
LGRID of
lgrid of
Lgri
```

- De volgende characters worden onderscheiden als scheidingstekens:

```
spatie
slash          /
is gelijk teken =
haakje openen (
haakje sluiten)
komma          ,
dubbele punt   :
punt komma     ;
```

- Sommige opdrachten moeten worden gevolgd door invoer. De volgende conventies worden gehanteerd voor het meegeven van de invoer:
 - Tussen de opdracht en de invoer bevindt zich een scheidingsteken.
 - Sommige opdrachten moeten worden gevolgd door een string, dit wordt als volgt aangegeven:

```
OPDRACHT <'string'> of bijvoorbeeld
OPDRACHT=<'string'>
```

Een string moet altijd tussen enkele quotes (' ') worden opgegeven. Het is mogelijk om in de string bepaalde characters op te nemen die normaal niet mogelijk zijn in een string, bijvoorbeeld de quote zelf. Dit kan met behulp van een zogenaamd "escape" character. In RSDS is dit de back-slash:

```
OPDRACHT=<\'\'string\'\'>
```

Bij sommige invoer strings wordt geen onderscheid gemaakt tussen hoofd- en kleine letters. Als er wel onderscheid wordt gemaakt, dan wordt dit bij de beschrijving van de opdracht vermeld.

- Sommige opdrachten moeten worden gevolgd door één of meer getallen (integers of reals, echter geen E-machten). Dit wordt als

volgt aangegeven:

OPDRACHT <val>	ingeval van één real
OPDRACHT <ival>	ingeval van één integer
OPDRACHT <val, val>	ingeval van twee reals, etc.

Een getal kan bestaan uit de volgende tekens:

teken van het getal	+ of -
numerieke waarden	0 1 2 3 4 5 6 7 8 9
decimale punt	

- Invoer kan over meerdere regels worden opgegeven. Als een opdracht moet worden gevolgd door invoer, en er wordt een return gegeven in plaats van invoer, dan kan dit nog steeds op de volgende regel.
- Het opgeven van opdrachten kan achter elkaar gebeuren indien de opdrachten gescheiden worden door een scheidingsteken. Een nieuwe opdracht hoeft dus niet op een nieuwe regel gegeven te worden.
- Het is mogelijk om in invoer files commentaar op te nemen met behulp van het "#" teken. Iedere regel die met dit teken begint, wordt als commentaar beschouwd.

2.3.3 Beschrijving

Deze sub-paragraaf geeft een beschrijving van de opdrachten die door RSDS uitgevoerd kunnen worden. Bij iedere opdracht wordt aangegeven welke invoer nodig is, wat de default waarde is en wanneer de opdracht geldig is. Een aantal opdrachten wordt toegelicht met behulp van een klein voorbeeld. Uitgebreidere voorbeelden zijn opgenomen in hoofdstuk 1.

De opdrachten zijn beschreven per categorie:

- variabele
- uitvoer algemeen
- uitvoer ascii
- uitvoer matlab
- opties

variabele

Instellen van variabele:**NAME <'STRING'>**

Het opgeven van de naam van de variabele. Dit is de naam van de variabele op de SDS file. Een overzicht van alle variabelen op de SDS file kan worden verkregen met behulp van de opdracht **TREE**. De variabele kan worden opgegeven door het invoeren van de naam:

```
name 'mmax'
```

of door het invoeren van de naam voorafgegaan door het pad:

```
name '/mesh/idimen/mmax'
```

Het is wel nodig dat de gebruiker de namen en paden van variabelen op de SDS file kent, bijvoorbeeld:

```
name '/mesh/idimen/kmax'      (aantal lagen)
name '/solution_flow/sep'     (waterstanden op hele
                               tijdstippen)
name '/solution_flow/up'      (u-snelheden op hele
                               tijdstippen)
```

Voor een beschrijving van alle variabelen op de SDS file wordt verwezen naar "Local data structure (LDS) WAQUA-in-SIMONA", [2], en naar hoofdstuk 7 waarin voorbeelden worden gegeven van de SDS documentatie files. Indien de naam van een variabele meer dan één keer op de LDS voorkomt (de paden voorafgaand aan de naam zijn verschillend), dan leest RSDS de variabele die het eerst in de LDS voorkomt.

Default: " (blanko, d.w.z. geen variabele ingesteld)
De opgegeven naam blijft geldig totdat een andere naam wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

LGRID

Als een variabele een lgrid index heeft, dat wil zeggen dat de lengte van de variabele gelijk is aan "mnmaxk", dan kan deze opdracht gegeven worden. Het geven van deze opdracht zorgt ervoor dat de lgrid-index van de SDS file gelezen kan worden.

Het geven van deze opdracht is nodig voor het wegschrijven van uitvoer. Lgrid variabelen kunnen als volledige matrices worden weggeschreven. In dit geval krijgen de waarden buiten de lgrid-index een dummy waarde.

Default: ' ', d.w.z. er wordt niet naar de lgrid-index op de SDS file verwezen. Het geven van de **LGRID** opdracht zorgt ervoor dat wordt verwezen naar de lgrid-index die op de SDS file wordt bewaard onder "/mesh/lgrid". Deze verwijzing blijft aan totdat de **LGRID** opdracht wordt uitgezet. Dit gebeurt door het opgeven van:

```
lgrid ''
```

of door het geven van de **DEFAULT** opdracht.
Het aanblijven van de **LGRID** opdracht is niet van invloed op variabelen die niet lgrid-geïndiceerd zijn.

**START <IVAL> OF
START <'STRING'>**

Opgeven van het begin adres van de variabele. Dit kan door het opgeven van een getal, bijvoorbeeld:

```
start 1      (begin van het array)
```

of door het opgeven van een variabele uit de LDS waarmee een getal wordt aangeduid, bijvoorbeeld:

```
start '+mnmaxk' (startwaarde + mnmaxk)
```

De variabele uit de LDS wordt opgegeven tussen quotes en kan vooraf worden gegaan door een "+" of een "*" teken.

In het voorgaande voorbeeld wordt bij het begin adres "mnmaxk" opgeteld. Dit is noodzakelijk voor het wegschrijven van waarden per laag. Alle waarden per laag liggen namelijk achter elkaar in één array: de eerste mnmaxk waarden voor de eerste laag, de tweede mnmaxk waarden voor de tweede laag, etc.

Het opgeven van het "+" teken in de string zorgt ervoor dat de nieuwe startwaarde mnmaxk wordt opgeteld bij de eerder opgegeven startwaarde 1. Het "+" teken mag echter alleen op de eerste positie van de string voorkomen en niet op een andere positie. Het geven van de opdracht:

```
start 'mnmaxk+mnmaxk'
```

geeft dan namelijk de foutmelding "mnmaxk+mnmaxk does not exist!", zie paragraaf 3.3.4.

Het opgeven van een harde waarde in de string of het opgeven van het "+" teken in combinatie met een harde waarde in de string is ook niet toegestaan, bijvoorbeeld:

```
start '+5'
```

geeft de melding "+5 does not exist!", zie paragraaf 2.3.4. Dit moet worden opgegeven door:

```
start 6      (startwaarde(=1) + 5)
```

Default: 1

De opgegeven start waarde blijft staan totdat een andere start waarde wordt opgegeven of totdat de **DEFAULT** opdracht wordt gegeven.

**LENGTH <IVAL> OF
LENGTH <'STRING'>**

Opgeven van een andere lengte van de variabele. Dit kan door het opgeven van een getal, bijvoorbeeld:

```
length 100
```

of door middel van een variabele uit de LDS, bijvoorbeeld:

```
length 'mmaxk'
```

De lengte "mmax*nmax" kan worden opgegeven in een string. Dit gaat als volgt:

```
length 'mmax'      (lengte wordt mmax)
length '*nmax'     (lengte wordt mmax * nmax)
```

Het opgeven van het "*" teken in de string zorgt ervoor dat de eerder opgegeven lengte "mmax" vermenigvuldigd wordt met de nieuwe lengte "nmax". Het "*" teken mag echter alleen op de eerste positie van de string voorkomen en niet op een andere positie. Het geven van de opdracht:

```
length 'mmax*nmax'
```

geeft namelijk de foutmelding "mmax*nmax does not exist!", zie paragraaf 2.3.4.

In plaats van het "*" teken kan ook het "+" teken worden opgegeven zoals bij de opdracht **START**.

Default: 0

Indien een variabele wordt ingesteld, dan wordt **LENGTH** de lengte van de variabele.

Een opgegeven lengte is geldig voor de opgegeven variabele. De lengte wijzigt indien: een andere lengte wordt opgegeven, een andere variabele wordt ingesteld, de **DEFAULT** opdracht wordt gegeven.

**DIMEN <IVAL1,
IVAL2>**

Opgeven van de dimensie van de variabele (2-dimensionaal).
ival1 = eerste dimensie, bijvoorbeeld mmax
ival2 = tweede dimensie, bijvoorbeeld nmax

Dit is nodig bij variabelen die niet de dimensie (mmax,nmax) hebben.
Opmerking lgrid variabelen hebben ook de dimensie (mmax,nmax),
zij hebben echter lengte mmaxk.

Het opgeven van de dimensie kan met behulp van getallen,
bijvoorbeeld:

```
dimen 201,173
```

Het opgeven van de dimensie (mmax,nmax) kan ook als volgt:

```
dimen 'mmax', 'nmax'
```

dus met behulp van variabelen uit de LDS. Het opgeven van:

```
dimen mmax,nmax
```

gaat echter niet goed! Dan volgt de melding dat de opdracht gevolgd
moet worden door een integer of een real, zie paragraaf 2.3.4.

Default: (<mmax>,<nmax>)

De opgegeven dimensie blijft staan totdat een andere dimensie wordt
opgegeven of totdat de **DEFAULT** opdracht wordt gegeven.

algemene uitvoer

Instellen van uitvoer (algemeen):**MNMN** <IVAL1,
IVAL2,
IVAL3,
IVAL4>

De begin- en eind-coördinaten van een veld voor het wegschrijven van de waarden van een variabele.

ival1 = minimum m waarde
ival2 = minimum n waarde
ival3 = maximum m waarde
ival4 = maximum n waarde

Het opgeven van een (deel)gebied kan met behulp van getallen. Bijvoorbeeld het deelgebied m=10,...,20 en n=15,...,30 wordt weggeschreven door:

```
mnmn 10,15,20,30
```

Het opgeven van een (deel)gebied kan ook met behulp van strings die een waarde aangeven. Bijvoorbeeld het hele gebied wordt weggeschreven door:

```
mnmn 1,1,'mmax','nmax'
```

Het opgeven van:

```
mnmn 1,1,mmax,nmax
```

gaat echter niet goed! Dan volgt twee keer een melding dat de opdracht gevolgd moet worden door een integer of een real, zie paragraaf 2.3.4.

Default: 1,1,<mmax>,<nmax> (het hele gebied)

Het opgegeven gebied blijft geldig tot een ander gebied wordt opgegeven of tot de opdracht **DEFAULT** wordt gegeven.

Opmerking Bij lgrid variabelen werkt de **MNMN** opdracht alleen als de **LGRID** opdracht gegeven is. Als deze opdracht niet gegeven is, dan wordt de uitvoer van de lgrid variabele voor het hele gebied weggeschreven. Bij niet-lgrid variabelen dienen de dimensies goed te zijn ingesteld, anders werkt de **MNMN** opdracht ook niet.

TIME <VAL, IVAL> OF
TIME '?'

Tijdstip(pen) waarop de waarden van een variabele gelezen en weggeschreven worden.

val = tijdstip (in minuten)

ival = code: 0 = variabele is tijdsafhankelijk of variabele is tijdsafhankelijk en eerste tijdstip moet nog gelezen worden

1 = het tijdstip dat wordt gelezen ligt met een nauwkeurigheid van DT / 1000 (DT = interval tussen tijdstippen) in de buurt van het opgegeven tijdstip

2 = het dichtsbijzijnde tijdstip (t.o.v. het opgegeven tijdstip) wordt gelezen

3 = het gelezen tijdstip is verkregen d.m.v. lineaire interpolatie tussen de tijdstippen die het dichtst bij het opgegeven tijdstip liggen

4 = ga naar volgend tijdstip en lees tijdstip

5 = ga naar vorig tijdstip en lees tijdstip

Het opgeven van een tijdstip is verplicht, het opgeven van een code is optioneel.

Bijvoorbeeld, indien waarden van een variabele gelezen moeten worden op het tijdstip 1440. (minuten), dan wordt het juiste tijdstip ingesteld door:

```
time 1440.
```

Het opgeven van een code wordt veel gebruikt in combinatie met de opdracht **REPEAT**. Als bijvoorbeeld de eerste 15 tijdstippen (beginnend vanaf het eerste tijdstip) moeten worden gelezen, dan kan dit opgegeven worden door:

```
time -1. 4
repeat 15
go
```

De (negatieve) waarde -1 in combinatie met de code 4 zorgt ervoor dat RSDS begint bij het eerste tijdstip. De herhaal factor 15 zorgt ervoor dat na het geven van de **GO** opdracht de eerste 15 tijdstippen gelezen worden.

Met behulp van de opdracht **TIME '?'** kunnen de tijdstippen worden opgevraagd waarop de waarden van een bepaalde variabele op de SDS file aanwezig zijn. Bijvoorbeeld:

```
name 'sep'
time '?'
```

geeft:

```
Available times:
.00 360.00 720.00 1080.00 etc.
```

Default: .00 (0) (tijdstip 0 met code 0).

Het ingevoerde tijdstip blijft staan totdat een ander tijdstip wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

Opmerking Indien de opdracht **TIME** wordt gegeven nadat een tijdsafhankelijke variabele is ingesteld, dan geeft RSDS de melding dat de variabele tijdsafhankelijk is.

TYPE 'I' OF 'R' OF 'C'

Opgeven van een type: integer ('i') of real ('r') of character ('c') voor het wegschrijven van de waarden van een variabele in de uitvoer. Deze opdracht kan worden gegeven als bijvoorbeeld reals als integers weggeschreven moeten worden of een integer als een character string. Als een ander type dan 'i', 'r' of 'c' wordt opgegeven, dan volgt een foutmelding, zie paragraaf 2.3.4

Default: " (geen type)

Indien een variabele wordt ingesteld, dan wordt **TYPE** het type van de variabele.

Een opgegeven type is geldig voor de opgegeven variabele. Het type wijzigt indien: een ander type wordt opgegeven, een andere variabele wordt ingesteld, de **DEFAULT** opdracht wordt gegeven.

REPEAT <VAL>

Herhaal factor voor het wegschrijven van de waarden van een variabele.
De uitvoer wordt <val> keer weggeschreven na de opdracht **GO**.
Deze opdracht kan worden gebruikt om waarden van een variabele op een aantal (opeenvolgende) tijdstippen achter elkaar weg te schrijven, zie de opdracht **TIME**.

Default: 1

De opgegeven herhaal factor geldt tot aan de **GO** opdracht. Daarna heeft de herhaal factor weer de waarde 1. De opgegeven herhaal factor wijzigt ook als een nieuwe herhaal factor wordt opgegeven of als de opdracht **DEFAULT** wordt gegeven.

DUMVAL <VAL>

Waarde waarmee dummy's in Igrid-structuur worden vervangen.
Een Igrid variabele kan als volledige (mmax,nmax) matrix worden weggeschreven. Voor de waarden die buiten de Igrid-index vallen worden dummy waarden weggeschreven.

Default: .00

De opgegeven dummy waarde blijft geldig totdat een nieuwe dummy waarde wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

GO

Het lezen en wegschrijven van de waarden van een variabele.
Na het geven van deze opdracht worden de waarden van een variabele volgens de opgegeven instellingen weggeschreven. Indien de naam van een uitvoer file is opgegeven, dan wordt de uitvoer naar die file geschreven, anders wordt de uitvoer naar het scherm geschreven. De uitvoer wordt naar de laatst opgegeven uitvoer file geschreven.
Als er geen variabele is ingesteld, dan geeft de **GO** opdracht een lege regel terug.

ascii uitvoer

Instellen van ascii uitvoer:**BOX**

Het aanzetten van de box-mode: de uitvoer wordt weggeschreven in box-formaat. Dit is het volgende formaat:

```
BOX MNMN=( <m1>, <n1>; <m2>, <m2>), VARIABLE_val=  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
BOX MNMN=( <m1>, <n3>; <m2>, <n4>), VARIABLE_val=  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000  
      .0000      .0000      .0000
```

etc.

Het aantal boxen en het aantal waarden per box is afhankelijk van de lengte van de variabele en het opgegeven format voor het wegschrijven van de uitvoer.

Default: True (de box-mode staat default aan)

De box-mode blijft aan totdat de matlab-mode wordt aangezet of totdat de opdracht **NOBOX** wordt gegeven.

NOBOX

Het uitzetten van de box-mode. De uitvoer wordt niet meer in box-formaat weggeschreven.

Default: False (de box-mode staat default aan)

De box-mode blijft uit totdat deze wordt aangezet of totdat de **DEFAULT** opdracht wordt gegeven.

MATLAB

Het aanzetten van de matlab-mode: de uitvoer wordt weggeschreven in matlab-formaat, d.w.z. in een script dat in MATLAB kan worden ingelezen en uitgevoerd. Met behulp van dit script worden de waarden van een variabele als volgt in matrices weggeschreven:

```
<naam variabele>( <m1>: <m2>, <n1>: <n2>) = [ ...
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000];
<naam variabele>( <m1>: <m2>, <n3>: <n4>) = [ ...
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000;
.0000          .0000          .0000];
etc.
```

Het aantal matrices per variabele en het aantal waarden per matrix is afhankelijk van de lengte van de variabele en het opgegeven format voor het wegschrijven van de uitvoer.

Default: False (de matlab-mode staat default uit)

De matlab-mode blijft aan totdat de box-mode wordt aangezet, of totdat de opdracht **NOMATLAB** wordt gegeven of totdat de opdracht **DEFAULT** wordt gegeven.

NOMATLAB

Het uitzetten van de matlab-mode: de uitvoer wordt niet in matlab-formaat weggeschreven.

Default: True (de matlab-mode staat default uit)

De matlab-mode blijft uit totdat deze wordt aangezet.

Opmerkingen

- Indien de box-mode en de matlab-mode beiden uit staan, dan wordt de uitvoer volgens het default format of volgens het opgegeven format weggeschreven.
- Bij Igrid variabelen werken de box-mode en de matlab-mode alleen indien de **LGRID** opdracht gegeven is. Als deze opdracht niet gegeven is, dan wordt de uitvoer volgens het gewone (default of opgegeven) format weggeschreven.
- Bij niet Igrid variabelen werken de box-mode en de matlab-mode alleen indien de dimensies van de variabele goed ingesteld zijn.
- Character arrays kunnen niet worden weggeschreven in box-formaat of matlab-formaat.
Dit kan worden opgelost door deze arrays in gewoon formaat weg te schrijven (**NOBOX** en **NOMATLAB**) of door de box- of matlab-formats na te bootsen, zie paragraaf 6.7.

PRINT <'STRING'>

Het wegschrijven van een opgegeven tekst. De opgegeven tekst wordt direct weggeschreven, zowel naar uitvoer file als naar standaard uitvoer (naar scherm). Bij het wegschrijven van de tekst wordt onderscheid gemaakt naar hoofd- en kleine letters.

Default: " (geen tekst)

Er wordt alleen tekst weggeschreven indien dit wordt opgegeven.

FORMAT <'STRING'>

Het opgeven van een format voor het wegschrijven van de waarden van variabelen in de uitvoer. Het format wordt opgegeven met behulp van "\$" tekens voor het aantal posities. Bijvoorbeeld: een integer weggeschreven met 4 posities wordt opgegeven door:

```
format '$$$$' (Fortran I4)
```

Een real met vier posities voor de decimale punt en met twee decimalen wordt opgegeven door:

```
format '$$$$.$$' (Fortran F7.2)
```

Het is mogelijk om een bepaald format te herhalen. Dit kan met behulp van de back-slash (voorafgegaan door het "escape" character, ook de back-slash), gevolgd door een herhaal factor en het format dat herhaald moet worden tussen haakjes, bijvoorbeeld:

```
format '\\3($$$.$ )'
```

geeft het format:

```
$$$.$ $$$.$ $$$.$
```

Let hierbij op de spatie voor het haakje sluiten. Als de spatie niet gegeven was, zou het format zijn:

```
$$$.$$$$$.$$$$$.$
```

Het is mogelijk om binnen één **FORMAT** opdracht verschillende formats op te geven, bijvoorbeeld:

```
format '\\3($$$ ) \\2($$$.$$ )
```

geeft het format:

```
$$$ $$$ $$$ $$$.$$ $$$.$$
```

Het is ook mogelijk om binnen een herhaald format nog één of meer herhaalde formats op te geven, bijvoorbeeld:

```
format '\\2(\\3($$$.$ ) )'
```

geeft het format, (let op de spaties):

```
$$$.$ $$$.$ $$$.$ $$$.$ $$$.$ $$$.$
```

Het is ook mogelijk om tekst in het format op te geven. Deze tekst wordt dan letterlijk weggeschreven (er wordt onderscheid gemaakt naar hoofd- en kleine letters), bijvoorbeeld:

```
format 'integer $$$'
```

geeft het format:

```
integer $$$
```

bij het geven van de **GO** opdracht wordt dan iedere waarde weggeschreven voorafgegaan door de tekst "integer".

Default: " (geen format)

Het opgeven format blijft staan totdat een nieuw format wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

PRITIM <'STRING'>

Het opgeven van een format voor het wegschrijven van tijden in de uitvoer. Deze opdracht zorgt ervoor dat na een **GO** opdracht het tijdstip van de ingestelde variabele wordt weggeschreven. Als een variabele tijdsafhankelijk is, of als er geen variabele is ingesteld, dan wordt geen tijdstip weggeschreven.

Het opgeven van het format gaat ook met behulp van "\$" tekens, zie de opdracht **FORMAT**. Omdat echter de waarde van één tijdstip wordt weggeschreven, hoeft slechts het format voor één waarde te worden opgegeven. Bijvoorbeeld:

```
time 1440.  
pritim '$$$$.'
```

geeft na het instellen van een tijdsafhankelijke variabele en na het geven van de **GO** opdracht als uitvoer:

```
1440.
```

en

```
time 2880.  
pritim '$$$$.$$'  
go
```

geeft:

```
2880.00
```

maar

```
time 14400.  
pritim '$$$$.'
```

geeft:

```
*****
```

omdat de waarde van het tijdstip (5 posities voor de decimaal) niet in het format past (4 posities voor de decimaal).

Het is mogelijk om tekst mee te geven, bijvoorbeeld:

```
time 2880.  
pritim 'Tijdstip $$$$. $$ (minuten)'  
go
```

geeft:

```
Tijdstip 2880.00 (minuten)
```

Default: " (geen format)

Het opgegeven format blijft staan totdat een nieuw format wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

OUTFILE <'STRING'>

Het opgeven van de naam van een ascii uitvoer file. Het is mogelijk om uitvoer naar verschillende uitvoer files te schrijven. Als deze opdracht gegeven is, dan wordt de uitvoer die vervolgens wordt gegenereerd naar de opgegeven uitvoer file geschreven. Als een nieuwe uitvoer file wordt opgegeven, dan wordt de vorige gesloten en wordt de uitvoer naar de nieuwe uitvoer file geschreven. Als de opdracht **MATFILE** wordt gegeven, dan wordt ook de ascii uitvoer file gesloten en wordt een nieuwe matlab uitvoer file geopend. Er kan dus naar maximaal één uitvoer file (ascii of binair) tegelijk worden geschreven. Bij het opgeven van de naam kan onderscheid worden gemaakt naar hoofd- en kleine letters.

Default: " (geen ascii uitvoer file, uitvoer wordt geschreven naar standaard uitvoer. Dit is naar scherm, of "redirected" naar de uitvoer file die tijdens het starten van RSDS is opgegeven, zie paragraaf 2.2).

De opgegeven naam van de ascii uitvoer file blijft staan totdat een nieuwe naam wordt opgegeven (van een ascii of een matlab uitvoer file) of totdat de opdracht **DEFAULT** wordt gegeven.

binaire matlab uitvoer**Instellen van binaire matlab uitvoer:****MATFILE <'STRING'>**

Het opgeven van de naam van de matlab uitvoer file. Indien deze opdracht wordt gegeven, wordt de uitvoer vervolgens binair en in matlab formaat weggeschreven. De vorige uitvoer file (ascii of matlab) wordt daartoe gesloten. Bij het opgeven van de naam kan onderscheid worden gemaakt naar hoofd- en kleine letters.

Default: " (geen matlab uitvoer file. Er wordt naar standaard uitvoer geschreven of naar een ascii uitvoer file). De opgegeven naam van de matlab uitvoer file blijft staan totdat een nieuwe naam wordt opgegeven (van een ascii of een matlab uitvoer file) of totdat de opdracht **DEFAULT** wordt gegeven.

MATNAME <'STRING'>

Het opgeven van de matlab naam van een variabele. Het is mogelijk om variabelen die een bepaalde naam hebben in de SDS file met een andere naam naar de matlab uitvoer file te schrijven. Dit geldt voor zowel ascii als binaire matlab uitvoer.

Bijvoorbeeld:

```
name 'sep'
matname 'waterlevel'
```

zorgt ervoor dat de variabele "sep" op de SDS file in de matlab uitvoer file met de naam "waterlevel" wordt weggeschreven.

Default: <'naam variabele op SDS file' (in hoofdletters)>

De opgegeven matlab naam blijft staan totdat een andere matlab naam wordt opgegeven, een andere variabele wordt ingesteld of de opdracht **DEFAULT** wordt gegeven.

MATTIME

Het wegschrijven van tijdreeksen in de matlab uitvoer file. De waarden van een variabele worden na de **GO** opdracht weggeschreven naar de matlab uitvoer file voor het opgegeven gebied (**MNMN**) en de opgegeven tijdstippen (**TIME**). De gegevens worden als volgt in matrices weggeschreven: de gegevens op een bepaald tijdstip worden in één kolom weggeschreven. De kolom begint met het tijdstip gevolgd door de waarden van de variabele. Als de variabele 2-dimensionaal is, dan worden de waarden 1-dimensionaal weggeschreven, waarbij de eerste index het snelst gaat. De uitvoer ziet er dan als volgt uit:

```
<naam> =
  [tijd 1          tijd 2      ...      tijd T
  waarde(1,1)      |           |           |
  waarde(2,1)      |           |           |
  waarde(3,1)      |           |           |
  ...              |           |           |
  waarde(nmax,1)   |           |           |
  waarde(1,2)      |           |           |
  ...              |           |           |
  waarde(nmax,2)   |           |           |
  ...              |           |           |
  ...              |           |           |
  waarde(nmax,mmax)|           |           | ];
```

Default: False (**MATTIME** staat default uit)

De opdracht blijft staan totdat opnieuw de opdracht **MATTIME** wordt gegeven of totdat de **DEFAULT** opdracht wordt gegeven.

Opmerkingen

Het wegschrijven van gegevens naar een binaire matlab file heeft als voordeel:

- dat tijdreeksuitvoer kan worden weggeschreven,
- dat het inlezen van een binaire matlab file in MATLAB ongeveer 100 keer sneller gaat dan het inlezen van een ascii MATLAB file.

Indien binaire matlab files overgehaald worden m.b.v. ftp, dient dit in de binaire mode te gebeuren.

opties**Instellen opties:****SHOW**

Het laten zien van de actuele instellingen. Na het geven van deze opdracht volgt een overzicht van de huidige instellingen. Hieronder volgt een overzicht van de default instellingen (opdracht **DEFAULT** gevolgd door de opdracht **SHOW**):

```
sdsfile      = SDS-<opc>
experiment   = <exp>
name         =
type         =
start        = 1
length       = 0
format       =
lgrid        =
dimen        = ( mmax, nmax)
mnmn         = ( 1, 1; mmax, nmax)
```



```

dumval      = .00E00
box         = T
matlab      = F
time       = .00 (0)
pritim     =
repeat     =
outp. file =
MAT file   =
MAT name   =
MAT time   = F

```

DEFAULT

Het initialiseren van de instellingen met behulp van default waarden. Na het geven van deze opdracht krijgen de instellingen hun default waarde. Zie de opdracht **SHOW** voor een overzicht van de instellingen met hun default waarden.

TREE <'STRING'> OF TREE '/'

Het laten zien van de paden en variabelen in de SDS file. Het geven van deze opdracht gevolgd door de naam van de variabele geeft als uitvoer het pad van de variabele in de SDS file. Bijvoorbeeld:

```
tree 'sep'
```

geeft:

```
Tree for /SOLUTION_FLOW/SEP:
```

Het geven van de **TREE** opdracht gevolgd door een pad, geeft een overzicht van de variabelen en paden onder het opgegeven pad. De variabelen worden weergegeven in kleine letters, de paden in hoofdletters. Bijvoorbeeld:

```
tree 'solution_flow'
```

geeft:

```
Tree for /SOLUTION_FLOW:
sep          up          vp
W           wphys
CZ
```

Het geven van de opdracht **TREE '/'** geeft een overzicht van alle paden en variabelen op de SDS file.

Standaard worden bij de **TREE** opdracht alle paden en variabelen op het eerste niveau (level) getoond. De paden en variabelen op onderliggende niveau's (levels) kunnen getoond worden met behulp van de opdracht **NLEVEL**.

NLEVEL <IVAL>

Het aantal levels dat getoont wordt m.b.v. de opdracht **TREE**. Als de gebruiker een overzicht wil van de paden en variabelen op onderliggende niveau's, dan kan dit door het geven van deze opdracht. Bijvoorbeeld, de paden en variabelen van de eerste twee niveau's worden getoond met behulp van de opdracht:

```
nlevel 2
```

Het geven van de opdracht:

```
tree 'solution_flow'
```

laat nu zien:

```
Tree for /SOLUTION_FLOW:
sep                up                vp
W
  w1                w2
  wphys
CZ
  czu                czv
```

Het laten zien van alle paden en variabelen op de eerste drie niveaus kan met behulp van de volgende opdrachten:

```
nlevel 3
tree '/'
```

Default: 1 (eerste niveau)

Het opgegeven niveau blijft geldig totdat een ander niveau wordt opgegeven of totdat de opdracht **DEFAULT** wordt gegeven.

ECHO

Het schrijven van de invoer op de command line naar standaard output. Er wordt als het ware een echo van de invoer gegeven.

Default: F (de echo staat default uit)

De echo blijft aan totdat opnieuw de opdracht **ECHO** wordt gegeven.

REOPEN

Het opnieuw openen van de SDS file. Deze optie wordt gebruikt om bijvoorbeeld met RSDS een SDS file te lezen, waarin door een ander programma (bijvoorbeeld WAQUA) nog gegevens weggeschreven worden. Het andere programma moet de SDS file dan wel in de "append-mode" geopend hebben (d.w.z. er worden gegevens toegevoegd). Als het andere programma de SDS file in "write-mode" geopend heeft, (d.w.z. er kunnen gegevens overschreven worden), dan stopt RSDS, zie ook paragraaf 4.2 Beperkingen van RSDS.

Na het geven van de **REOPEN** opdracht kunnen de nieuwe gegevens, die inmiddels door het andere programma aan de SDS file zijn toegevoegd, gelezen worden.

EXIT

Het verlaten van RSDS. Na het geven van deze opdracht sluit RSDS alle uitvoer files en wordt RSDS verlaten. Indien de RSDS opdrachten worden gegeven door middel van een invoer file, dan hoeft deze opdracht niet

gegeven te worden. RSDS stopt zodra het einde van de invoer file bereikt is.

2.3.4 Foutmeldingen

Indien een opdracht verkeerd wordt ingevoerd of indien de invoer van een opdracht niet aan de in paragraaf 2.3.2 beschreven conventies voldoet, dan kunnen foutmeldingen of gewone meldingen volgen. Indien foutmeldingen volgen, dan worden deze geteld. Als er tien meldingen geconstateerd zijn, dan stopt RSDS. In deze subparagraaf worden de foutmeldingen en de gewone meldingen van RSDS beschreven.

foutmeldingen

Indien een opdracht verkeerd wordt ingevoerd (bijv. een typefout) of indien een opdracht wordt ingevoerd die niet bestaat, dan volgt de melding:

```
*****
*** ERROR number 2701 ,
    in module RSDS: rsdcon

    Unrecognizable keyword used in the input:
    <opdracht>

*****
```

Indien een opdracht gevolgd moet worden door een 'string', dan wordt een foutmelding gegeven indien de opdracht niet gevolgd is door een 'string':

```
*****
*** ERROR number 2702 ,
    in module RSDS :rsdcon

    Keyword <opdracht> not followed by a
    quoted string.

*****
```

Als per ongeluk na zo'n opdracht een return is gegeven in plaats van een string, kan deze nog op de volgende regel worden ingevoerd. Wordt echter op de volgende regel iets anders ingevoerd, bijvoorbeeld nog een keer dezelfde opdracht of een andere opdracht, dan ziet RSDS dit als de gevraagde invoer en volgt weer een foutmelding.

Indien opgegeven invoer niet vooraf wordt gegaan door een opdracht, dan volgt de foutmelding:

```
*****
*** ERROR number 2705 ,
    in module RSDS :rsdcon

    Keyword missing in input text.

*****
```

Deze foutmelding kan een gevolg zijn van een vorige foutmelding, bijvoorbeeld het invoeren van:

```
name <return>
name <'string'>
```

geeft:

```
*****

*** ERROR number 2702 ,
    in module RSDS   :rsdcon

    Keyword <opdracht> not followed by a
    quoted string.

*****
```

omdat de tweede opdracht **NAME** wordt gezien als invoer van de eerste opdracht **NAME**. De invoer <'string'> wordt nu als invoer gezien zonder voorafgaande opdracht. Daardoor volgt:

```
*****

*** ERROR number 2705 ,
    in module RSDS   :rsdcon

    Keyword missing in input text.

*****
```

meldingen

Indien een opdracht gevolgd moet worden door een waarde, bijvoorbeeld een integer of een real, dan wordt een melding gegeven als de opdracht niet gevolgd wordt door de gevraagde waarde:

```
RSDS:rsdcon: keyword <opdracht>
    should be followed by an
    integer/real/character
```

Indien een opdracht gevolgd moet worden door twee of meer waarden, dan wordt deze opdracht net zolang gegeven totdat het juiste aantal waarden is ingevoerd. Dit kan tot andere (fout)meldingen leiden. Bijvoorbeeld bij de opdracht **MNMN** moeten vier integers volgen. Als er twee waarden zijn opgegeven in plaats van vier, dan volgt eerst de bovenstaande melding. Als vervolgens nog een waarde wordt ingevoerd gevolgd door een andere opdracht (in plaats van nog twee waarden), dan volgt weer bovenstaande melding (plus eventueel andere (fout)meldingen als gevolg van de ingetikte opdracht), net zolang tot het juiste aantal waarden is ingevoerd.

Het opgeven van een verkeerd type waarde (bijvoorbeeld een integer in plaats van een real of andersom) leidt niet tot een melding. In dit geval wordt de ingevoerde waarde door RSDS geconverteerd naar het gevraagde type.

Aan sommige opdrachten moet een LDS variabele worden meegegeven, bijvoorbeeld de naam van een variabele of de waarden van tijdstippen op de SDS file. Als bij zo'n opdracht een niet bestaande LDS variabele wordt opgegeven, dan volgt de melding:

```
<'string'> does not exist !
```

Deze melding wordt ook gegeven bij opdrachten die gevolgd worden door een 'string', terwijl de opdracht eigenlijk gevolgd moet worden door een waarde (integer of real).

Bij de opdracht **TYPE** kan alleen worden opgegeven: 'i' (voor integer), 'r' (voor real) of 'c' (voor character). Indien een ander of een niet bestaand type wordt opgegeven, dan volgt de melding:

```
RSDS      :rsdcon: invalid TYPE value
```

Samenvatting:

foutmeldingen:

- Unrecognizable keyword used in the input
- Keyword not followed by a quoted string
- Keyword missing in input text

meldingen:

- Keywordt should be followed by an integer/real/character
- <'string'> does not exist
- invalid TYPE value

2.4 RSDS in batch

Het is mogelijk om RSDS in batch te starten. Door dit te doen kan RSDS in de achtergrond gedraaid worden. Het starten van RSDS in batch kan op twee manieren:

- batch file**
- batch commando's**
 - met behulp van een batch file
 - door het geven van batch commando's op de command line

Deze twee mogelijkheden worden in de volgende sub-paragrafen besproken.

2.4.1 Met behulp van een batch file

batch file Zo'n batch file bevat de opdracht rds.pl en een aantal opties:

```
rds.pl  -runid  <opc>          \
        -infile <invoer>       \
        -outfile <uitvoer>    \
        -back  <y of n>
```

De regels in zo'n batch file worden met elkaar verbonden (behalve de laatste regel) met behulp van de back-slash.

De batch file wordt uitgevoerd door het intikken van de naam van de batch file op de command line:

```
<naam batch file>  bijvoorbeeld
start.rds
```

starten rds De opdracht rds.pl start RSDS. De opties zorgen voor het volgende:

run id - De optie -runid <opc> geeft de run identificatie (dit is de opc van de SDS file) aan.

invoer - De optie -infile <invoer> geeft de invoer aan. Indien de RSDS opdrachten zijn opgegeven in een invoer file, dan wordt de naam van de invoer file meegegeven. Als de invoer handmatig wordt opgegeven, dan moet achter -infile het "-" teken worden ingevoerd:

```
-infile -
```

Als achter -infile niets wordt opgegeven, dan wordt de volgende optie -outfile als de naam van de invoer file gezien. Dit levert foutmeldingen op.

uitvoer - De optie -outfile <uitvoer> geeft de uitvoer aan. Als de uitvoer naar een file moet worden geschreven, dan wordt de naam van de uitvoer file opgegeven. Als er standaard uitvoer moet worden weggeschreven, dan moet worden opgegeven:

```
-outfile -
```

background - De optie -back geeft aan of RSDS in de achtergrond wordt uitgevoerd (y) of in de voorgrond (n).

Een optie kan worden weggelaten in de batch file. RSDS vraagt hier dan alsnog om in de voorgrond.

Indien een optie wel wordt opgegeven, maar niet door de benodigde tekst wordt gevolgd, dan wordt de optie op de volgende regel als de benodigde tekst gezien. Dit kan tot foutmeldingen leiden.

foutmeldingen

Voor de opties -runid, -infile en -outfile gelden dezelfde foutmeldingen zoals beschreven in paragraaf 2.1 Starten van RSDS (op de voorgrond).

2.4.2 Met behulp van batch commando's

batch commando's

In plaats van het uitvoeren van een batch file kunnen de batch commando's ook direct op de command line worden ingevoerd:

```
rsds.pl -runid <opc> -infile <invoer> -outfile <uitvoer> - back <y  
of n>
```

Het opgeven van de opties gaat op dezelfde manier zoals beschreven voor de batch file (paragraaf 2.4.1).

Voor de foutmeldingen wordt verwezen naar paragraaf 2.1 Starten van RSDS (op de voorgrond).

3 TECHNISCHE ACHTERGROND RSDS

RSDS kan worden toegepast op iedere SDS file waarvan een zogenaamde SDS documentatie file bestaat. Dit wordt toegelicht in paragraaf 3.1.

RSDS is gebaseerd op SIMONA tools en routines en op het directory systeem DIRSYS. Voor de SIMONA tools en routines wordt verwezen naar "Programmer's guide SIMONA", [1]. In paragraaf 3.2 wordt een korte toelichting gegeven op DIRSYS.

3.1 De SDS documentatie file

RSDS kan worden gebruikt voor het lezen van gegevens van alle SDS files waarvoor de LDS is vastgelegd in een SDS documentatie file.

locale data structuur

Zo'n SDS documentatie file bevat de locale data structuur (LDS) van alle gegevens op de SDS file: de namen van compound arrays, de namen, typen en lengtes van data elementen (arrays of enkelvoudige variabelen). Deze informatie is nodig voor het lezen van gegevens van de SDS file.

De volgende programma's leveren SDS files waarvoor de LDS reeds is vastgelegd in SDS documentatie files:

<u>Programma</u>	<u>SDS documentatie file(s)</u>
WAQUA	sds_WAQUA.doc
WAQAD	sds_WAQAD.doc, sds_WAQUA.doc, sds_OBSERVED.doc
KALMAN	sds_KALMAN.doc, sds_WAQUA.doc, sds_OBSERVED.doc
WAQWND	sds_WIND.doc

Het programma WAQUA wordt gebruikt voor de berekening van twee- en drie-dimensionale waterbeweging.

Met behulp van het programma WAQAD kunnen twee- en drie-dimensionale WAQUA-modellen afgeregeld worden.

Het programma KALMAN berekent een "steady state Kalman filter" voor het filteren van met WAQUA berekende waterstanden.

Het programma WAQWND leest gegevens uit een KNMI wind file en schrijft deze gegevens naar een wind SDS file.

Omdat zowel WAQAD als KALMAN meetgegevens in de berekeningen verwerken, zijn beide SDS files van deze programma's ook gebaseerd op de SDS documentatie file voor meetwaarden (sds_OBSERVED.doc). Ook zijn de SDS files van beide programma's gebaseerd op de SDS documentatie file van WAQUA, aangezien beide programma's WAQUA berekeningen uitvoeren.

Voor een beschrijving van de bovenstaande programma's wordt verwezen naar de betreffende User's guides. Voorbeelden van bovenstaande SDS documentatie files zijn opgenomen in hoofdstuk 7.

Opmerkingen

- De run-procedure "rsds.pl" voegt tijdens het uitvoeren van RSDS alle in deze paragraaf genoemde SDS documentatie files tijdelijk samen in de file "lds.dlds".
- Voor het lezen van gegevens van een wind SDS file is alleen de documentatie file sds_WIND.doc nodig. Omdat de run-procedure "rsds.pl" echter alle SDS documentatie files samenvoegt, is een aparte run-procedure geschreven die alleen de SDS documentatie file voor windgegevens gebruikt. Deze procedure heet "rsds_wind.pl" en bevindt zich op het rekencluster "compcl" op de directory:

/users/applic/waqad/bin

De reden voor het schrijven van deze aparte procedure is dat de wind LDS en de overige LDS-en kunnen conflicteren.

- Het is mogelijk om met RSDS een SDS file te lezen die gebaseerd is op een eigen (niet in deze paragraaf genoemde) SDS documentatie file. Daartoe kopieert de gebruiker zijn eigen SDS documentatie file naar de file "lds.dlds" op de directory waar RSDS uitgevoerd wordt. RSDS gebruikt dan deze eigen documentatie file.

3.2 Het directory systeem DIRSYS

directory systeem

Het directory systeem DIRSYS bestaat uit libraries en routines die het mogelijk maken voor de programmeur om toegang te krijgen tot de SIMONA LDS door het opgeven van data namen. De conventies voor het opgeven van deze data namen corresponderen met de directory- en file-structuur in UNIX.

DIRSYS is gebaseerd op de SIMONA toolbox en maakt tevens gebruik van karakteristieke SIMONA mechanismen.

Met behulp van DIRSYS kan informatie worden opgevraagd uit SDS documentatie files en kunnen gegevens uit de SDS files worden gelezen.

Zie voor meer informatie "User's guide WAQAD", [3], en "System documentation WAQAD", [4].

4 GEWENSTE UITBREIDINGEN EN BEPERKINGEN

Het programma RSDS is al enige jaren in gebruik bij een groeiend aantal personen. Dit heeft geleid tot aanvullende wensen, die tot op heden (nog) niet gerealiseerd zijn. In dit hoofdstuk worden deze gewenste uitbreidingen besproken (paragraaf 4.1). Ook worden de bekende beperkingen van het programma besproken (paragraaf 4.2).

4.1 Gewenste uitbreidingen

De volgende uitbreidingen van RSDS zijn gewenst:

- **Wijzigingen in de SIMONA toolbox.**
Ten behoeve van RSDS zijn in twee standaard SIMONA routines wijzigingen aangebracht. Deze wijzigingen zijn niet opgenomen in de standaard SIMONA toolbox bibliotheek. Teneinde conflict situaties in de toekomst te voorkomen is het wenselijk deze aanpassingen in de standaard bibliotheek op te nemen.
- **De dimensie van arrays.**
In de huidige versie van RSDS heeft iedere variabele een 1-dimensionale lengte, gespecificeerd in de SDS documentatie file. Het programma RSDS kan maximaal een 2-dimensionaal array verwerken. Het is aan te bevelen hierin wijzigingen aan te brengen, zodat voor ieder variabele de dimensie opgegeven kan worden en de grootte van deze dimensies in de SDS documentatie file. RSDS zou dan ook meer-dimensionale arrays moeten kunnen verwerken.
- **Het opslaan van de data structuur.**
Op een SDS file worden gegevens opgeslagen via de LDS (Locale Data Structuur). Deze gegevens zijn voorzien van karakteristieke namen. Aan deze namen is de data structuur per definitie niet af te lezen, en zijn ook eventuele sub-namen binnen een data array niet bekend. RSDS heeft deze gegevens echter nodig, en leest deze gegevens uit de SDS documentatie file. Het zou zeer wenselijk zijn deze gegevens bij een experiment op te slaan. Ieder experiment kan immers een andere opbouw hebben, wat in de huidige versie van RSDS is opgelost door met verschillende documentatie files te werken. Bovendien kan dan bij wijziging van de LDS de "oude" LDS nog probleemloos ingelezen worden.
- **De opdracht INCR voor het opgeven van tijd-incrementen.**
Met behulp van deze opdracht kan een tijdsinterval worden opgegeven voor het lezen van de tijdstippen op de SDS file. Op deze manier hoeven niet alle (achtereenvolgende) tijdstippen gelezen te worden.

4.2 Beperkingen van RSDS

De volgende beperkingen van RSDS zijn bekend:

- Indien een ander programma een SDS file in "write-mode" geopend heeft, dan mag deze SDS file niet worden gelezen met RSDS. Als dit

wel gebeurt, dan stopt RSDS. Als een ander programma een SDS file in de "append-mode" of in de "read-only-mode" heeft geopend, dan kan de SDS file wel gelezen worden met RSDS.

Als een ander programma een SDS file, die reeds door RSDS wordt gelezen, in "write-mode" wil openen, dan stopt dit programma.

Opmerkingen

- "write-mode" = gegevens kunnen overschreven worden (in de "Programmers guide SIMONA" wordt dit de "update-mode" genoemd)
 - "append-mode" = gegevens kunnen toegevoegd worden
 - "read-only-mode" = gegevens kunnen alleen gelezen worden
 - WAQUA opent een SDS file in "append-mode".
Het proces "adpro" van WAQAD opent een SDS file in "write-mode". Als dit proces bezig is, dan kan de SDS file niet gelezen worden met RSDS.
 - Hoewel RSDS de SDS file in "read-only-mode" opent, dient de te lezen SDS file wel "write-permissie" te hebben. Als dit niet het geval is, dan stopt RSDS.
-
- Character arrays kunnen met RSDS niet als matlab ascii uitvoer worden weggeschreven.

5 REFERENTIES

- [1] *Programmer's guide SIMONA*, EDS, SIMONA report 90-09, 15 april 1994.
- [2] *Local data structure (LDS) WAQUA-in-SIMONA*, EDS, SIMONA report 92-09, 6 juli 1994.
- [3] *User's guide WAQAD*, SIMTECH, versie 2.02, 1 april 1996.
- [4] *System documentation WAQAD*, SIMTECH, versie 2.1, 25 maart 1996.
- [5] *User's guide KALMAN*, SIMTECH, versie 1.01, 17 oktober 1996.
- [6] *System documentation KALMAN*, SIMTECH, versie 1.0, 31 mei 1996.
- [7] *User's guide WAQUA,,* EDS, versie 8.20, 24 oktober 1996.

6 VOORBEELDEN INVOER RSDS

In dit hoofdstuk zijn voorbeelden opgenomen van RSDS invoer. Een aantal van de voorbeelden wordt toegelicht met behulp van uitvoer. De voorbeelden die zijn opgenomen in dit hoofdstuk bevinden zich ook op diskette en op het rekencluster "compcl" op de directory:

```
/users/applic/waqad/examples
```

De gebruiker kan deze voorbeelden direct gebruiken voor RSDS op een HP-9000 machine en op een PC.

Hieronder volgt een overzicht van de voorbeelden:

<u>Hoofdstuk</u>	<u>Voorbeeld</u>	<u>Omschrijving</u>
6.1	dimen.inp	het bekijken en instellen van dimensies
6.2	formats.inp	het instellen van formats voor het wegschrijven van ascii uitvoer
6.3	binair.inp	het wegschrijven van binaire matlab uitvoer
6.4	bndout.inp	het wegschrijven van boundary outlines
6.5	cco.inp	het wegschrijven van gridcel-afmetingen en x-y coördinaten
6.6	lagen.inp	het wegschrijven van gegevens per laag
6.7	stations.inp	het wegschrijven van stationsnamen
6.8	tijden.inp	het wegschrijven van tijdreeks-uitvoer

In dit hoofdstuk zijn ook voorbeelden van run-scripts opgenomen. Run-scripts zijn UNIX shell scripts waarmee RSDS meerdere keren achter elkaar uitgevoerd kan worden, waarbij steeds verschillende invoer wordt meegegeven. Hieronder volgt een overzicht van de run-scripts:

<u>Hoofdstuk</u>	<u>Run-script</u>	<u>Omschrijving</u>
6.9	rsds.gain	het lezen en schrijven van de Kalman gain per station
6.10	rsds.tijd	het lezen en wegschrijven van de waarden van een variabele op alle tijdstippen naar verschillende uitvoer files

Deze run-scripts zijn voor de gevorderde RSDS gebruiker! De run-scripts zijn ook beschikbaar op diskette en het rekencluster.

6.1 Voorbeeld: dimen.inp

Dit voorbeeld is weergegeven als handmatige invoer voor RSDS. De weergegeven opdrachten zijn opgenomen in de invoer file “dimen.inp”.

Eerst gaan we na wat de dimensies van het model zijn. Dit zijn:

```
SI> name='kmax' go      (aantal lagen)
      5
SI> name='mmax' go     (aantal m-gridpunten)
      201
SI> name='hmax' go     (aantal n-gridpunten)
      173
SI> name='mnmaxk' go   (aantal rekenpunten)
      19836
```

Vervolgens stellen we een variabele in, bijvoorbeeld de waterstand in rekenpunten. Deze variabele heeft op de SDS file de naam “sep”:

```
SI> name='sep'
```

We bekijken de instellingen van deze variabele met behulp van de opdracht show:

```
SI> show

sdsfile      = SDS-3dw
experiment   = feb93
name         = /SOLUTION_FLOW/SEP
type        = REAL
start       =      1
length      =   19836
format      = $$$$$$$$$$.$$$$ $$$$$$$$$$.$$$$
             $$$$$$$$$$.$$$$ $$$$$$$$$$.$$$$
             $$$$$$$$$$.$$$$

lgrid       =
dimen       = ( 201, 173)
mnmn       = (   1,   1; 201, 173)
dumval      =   .00E+00
box         = T
matlab      = F
time        =      .00 (0)
pritim     =
repeat      =      1
outp.file   =
MAT file    =
MAT name    = SEP
mattime     = F
```

We zien dat de lengte van de variabele gelijk is aan mnmaxk. Dit betekent dat variabele “sep” een lgrid index heeft. Daartoe geven we de opdracht:

```
SI> lgrid
```

Vervolgens bekijken we de instellingen van de waterstand in de waterstand-stations. Deze variabele heet “zwl”:


```

SI> name 'zwl'
SI> show

sdsfile      = SDS-3dw
experiment=  feb93
name         = /TIMEHISTORIES_FLOW/TIMHIS/ZWL
type         = REAL
start        =          1
length       =         159
format       = $$$$$$$$$.$$$$ $$$$$$$$$.$$$$
$$$$$$$$$.$$$$ $$$$$$$$$.$$$$ $$$$$$$$$.$$$$
lgrid        = /MESH/LGRID
dimen        = ( 201, 173)
mnmn         = (   1,   1; 201, 173)
dumval       =   .00E+00
box          = T
matlab       = F
time         =         .00 (0)
pritim       =
repeat       =          1
outp.file    =
MAT file     =
MAT name     = ZWL
mattime      = F

```

De instellingen laten zien dat variabele “zwl” lengte 159 heeft. Dit zijn het aantal waterstand-stations (variabele “nowl”, dit kan worden nagegaan in de SDS documentatie file):

```

SI> name 'nowl' go
      159

```

Het pad van de variabele “nowl” is:

```

SI> tree 'nowl'
Tree for /CONTROL_FLOW/ICONTA/NOWL

```

We stellen variabele “zwl” opnieuw in waarbij we de dimensie van deze variabele als volgt aanpassen:

```

SI> name 'zwl'
SI> dimen 1, '/control_flow/iconta/nowl'
# is gelijk aan: dimen 1,159
# en gelijk aan: dimen 1,'nowl'

SI> show

sdsfile      = SDS-3dw
experiment=  feb93
name         = /TIMEHISTORIES_FLOW/TIMHIS/ZWL
type         = REAL
start        =          1
length       =         159
format       = $$$$$$$$$.$$$$ $$$$$$$$$.$$$$
$$$$$$$$$.$$$$ $$$$$$$$$.$$$$ $$$$$$$$$.$$$$
lgrid        = /MESH/LGRID
dimen        = (   1, 159)
mnmn         = (   1,   1;   1, 159)
dumval       =   .00E+00
box          = T

```

```

matlab      = F
time        =          .00  (0)
pritim      =
repeat      =          1
outp.file   =
MAT file    =
MAT name    = ZWL
mattime     = F

```

Vervolgens willen we de lgrid-index zelf bekijken. De naam van deze variabele is "lgrid":

```

SI> name 'lgrid'
SI> show

```

```

sdsfile     = SDS-3dw
experiment=  feb93
name        = /MESH/LGRID
type        = INT
start       =          1
length      =   34773
format      = $$$$$$$$$$$$$$ $$$$$$$$$$$$$$
$$$$$$$$$$$$$$$$ $$$$$$$$$$$$$$$$$ $$$$$$$$$$$$$$$$$
lgrid       = /MESH/LGRID
dimen       = ( 201, 173)
mnmn       = (   1,   1; 201, 173)
dumval      =   .00E+00
box         = T
matlab      = F
time        =          .00  (0)
pritim      =
repeat      =          1
outp.file   =
MAT file    =
MAT name    = LGRID
mattime     = F

```

We zien dat de variabele "lgrid" zelf een volledige index heeft, want length = mmax*nmax. Er is echter een addertje onder het gras! De variabele "lgrid" heeft namelijk dimensies (nmax,mmax) i.p.v. (mmax,nmax). Daartoe wijzigen we de dimensies:

```

SI> dimen 'nmax','mmax'

```

Indien vervolgens een andere variabele wordt ingesteld, dan moet de dimensie van deze variabele wel opnieuw ingesteld worden.

6.2 Voorbeeld: formats.inp

We gaan de gridcelafmeting in m-richting (variabele "guu") met verschillende uitvoer formats wegschrijven naar ascii uitvoer files. De opdrachten, gegeven in dit voorbeeld, zijn opgenomen in de invoer file "formats.inp".

Variabele "guu" heeft een lgrid index. De variabele wordt als volgt ingesteld:

```
SI> name 'guu'
SI> lgrid
```

De uitvoer wordt standaard in box-formaat weggeschreven. Het wegschrijven van de waarden in het deelgebied (10,10;12,12) gaat als volgt:

```
SI> mnmn 10,10,12,12
SI> go
BOX MNMN=( 10, 10; 12, 12), VARIABLE_val=
    9265.8730    9265.8730    9265.8730
    9265.8730    9265.8730    9265.8730
    9265.8730    9265.8730    9265.8730
```

We veranderen het format en schrijven opnieuw de waarden weg:

```
SI> format '\\3($$$$$. $ )'
SI> go
BOX MNMN=( 10, 10; 12, 12), VARIABLE_val=
    9265.9    9265.9    9265.9
    9265.9    9265.9    9265.9
    9265.9    9265.9    9265.9
```

Vervolgens gaan we matlab uitvoer wegschrijven voor het deelgebied (115,115;120,120):

```
SI> matlab
SI> mnmn 115,115,120,120
SI> go
GUU( 115: 120, 115: 117) = [ ...
    7510.8    7964.1    8926.6;
    7493.8    7947.5    8911.1;
    7476.7    7931.0    8895.6;
    7459.7    7914.3    8880.0;
    7442.6    7897.7    8864.5;
    7425.5    7881.1    8848.9];
GUU( 115: 120, 118: 120) = [ ...
    6976.0    7425.5    7881.1;
    6958.5    7408.4    7864.4;
    6941.0    7391.3    7847.7;
    6923.5    7374.2    7831.0;
    6906.0    7357.0    7814.3;
    6888.4    7339.9    7797.6];
```

Daarna schrijven gewone uitvoer weg (niet in box- of matlab-formaat), voorafgegaan door een tekst:

```
SI> nomatlab
```

```
SI> mmmn 115,115,117,120
SI> print 'GUU heeft in deelgebied
(115,115;117,120)      de waarden:'
GUU heeft in deelgebied (115,115;117,120) de
waarden:
SI> go
7510.8    7964.1    8926.6
7493.8    7947.5    8911.1
7476.7    7931.0    8895.6
6976.0    7425.5    7881.1
6958.5    7408.4    7864.4
6941.0    7391.3    7847.7
```

6.3 Voorbeeld: binair.inp

In plaats van ascii matlab-uitvoer kan de uitvoer ook naar een binaire matlab file (matfile) worden geschreven. Zo'n matfile kan ingelezen worden in MATLAB. Dit gaat ongeveer 100 keer sneller dan het inlezen van een ascii matlab file.

Met behulp van de volgende invoer file wordt de diepte (variabele "h") naar de binaire matlab file "depth.mat" weggeschreven. In deze matfile heeft de diepte ook de naam "diepte" meegekregen in plaats van de naam "h".

```
# =====  
#   uitvoer binair = depth.mat  
# =====  
lgrid  
matfile='depth.mat'  
  
#   variabele H = diepte  
name='h'  
matname='diepte'  
go
```

6.4 Voorbeeld: bndout.inp

Het volgende voorbeeld bevat het wegschrijven van variabelen voor de boundary outlines. Dit zijn: de lengtes van de boundary outlines (variabele "land"), de x- en y-coördinaten van de boundary outlines (variabelen "xland" en "yland"). Deze variabelen hebben geen lgrid-index. De dimensies van deze variabelen moeten dan ook ingesteld worden.

Eerst worden deze variabelen weggeschreven in één ascii matlab file. Vervolgens worden de variabelen naar één binaire matlab file geschreven.

```
# =====  
#   uitvoer ascii   = bndout.m  
# =====  
matlab  
outfile='bndout.m'  
  
#   variabele LAND = lengte boundary outlines  
name='land'  
dimen=1,'/displays/iland/leng'  
go  
  
#   variabele XLAND = x-coor boundary outlines  
name='xland'  
dimen=1,'/displays/iland/nopt'  
# is gelijk aan dimen=1,'nopt'  
go  
  
#   variabele YLAND = y-coor boundary outlines  
name='yland'  
dimen=1,'/displays/iland/nopt'  
go  
  
# =====  
#   uitvoer binair  = bndout.mat  
# =====  
matfile='bndout.mat'  
  
#   variabele LAND = lengte boundary outlines  
name='land'  
go  
  
#   variabele XLAND = x-coor boundary outlines  
name='xland'  
go  
  
#   variabele YLAND = y-coor boundary outlines  
name='yland'  
go
```

6.5 Voorbeeld: cco.inp

In het volgende voorbeeld worden gridcel-afmetingen (variabelen “guu” en “gvv”) en x-y coördinaten (variabelen “xzeta” en “yzeta”) geschreven naar ascii en binaire matlab files. Iedere variabele wordt in een aparte uitvoer file weggeschreven. Deze variabelen hebben lgrid-index.

```
# =====
#   uitvoer ascii   = guu.m
#                   gvv.m
#                   xzeta.m
#                   yzeta.m
# =====
lgrid
matlab

#   variabele GUU = afmeting gridcel in m-richting
outfile='guu.m'
name='guu'
go

#   variabele GVV = afmeting gridcel in n-richting
outfile='gvv.m'
name='gvv'
go

#   variabele XZETA = x-coor in waterstandspunt
outfile='xzeta.m'
name='xzeta'
go

#   variabele YZETA = y-coor in waterstandspunt
outfile='yzeta.m'
name='yzeta'
go

# =====
#   uitvoer binair  = guu.mat
#                   gvv.mat
#                   xzeta.mat
#                   yzeta.mat
# =====
# lgrid staat nog aan

#   variabele GUU = afmeting gridcel in m-richting
matfile='guu.mat'
name='guu'
go
#   variabele GVV = afmeting gridcel in n-richting
matfile='gvv.mat'
name='gvv'
go

#   variabele XZETA = x-coor in waterstandspunt
```

```
matfile='xzeta.mat'  
name='xzeta'  
go
```

```
#   variabele YZETA = y-coor in waterstandspunt  
matfile='yzeta.mat'  
name='yzeta'  
go
```


6.6 Voorbeeld: lagen.inp

In het volgende voorbeeld worden gegevens per laag weggeschreven voor de u-snelheid in rekenpunten (variabele "up"). Deze variabele heeft lgrid-index.

Eerst wordt de uitvoer per laag naar aparte ascii matlab files geschreven. De uitvoer per laag wordt ook naar een binaire matlab uitvoer file geschreven. In dit geval worden de gegevens per laag naar één file geschreven waarbij verschillende namen voor de variabele worden meegegeven.

```
# =====
#   uitvoer ascii   = u_l1.m (matlab formaat)
#                   u_l2.m
#                   u_l3.m
#                   u_l3.m
#                   u_l4.m
#                   u_l5.m
# =====

#   variabele UP = u-snelheid in rekenpunten
name='up'
lgrid
matlab

# laag 1
outfile='u_l1.m'
go

# laag 2
outfile='u_l2.m'
start='+mnmaxk'
go

# laag 3
outfile='u_l3.m'
start='+mnmaxk'
go

# laag 4
outfile='u_l4.m'
start='+mnmaxk'
go

# laag 5
outfile='u_l5.m'
start='+mnmaxk'
go

# =====
#   uitvoer binair  = u_lagen.mat
# =====

#   variabele UP = u-snelheid in rekenpunten
name='up'
matfile='u_lagen.mat'
```

```
# laag 1
matname='u_l1'
# start staat nog op laag 5
start 1
go

# laag 2
matname='u_l2'
start='+mnmaxk'
go

# laag 3
matname='u_l3'
start='+mnmaxk'
go

# laag 4
matname='u_l4'
start='+mnmaxk'
go

# laag 5
matname='u_l5'
start='+mnmaxk'
go
```

6.7 Voorbeeld: stations.inp

In het onderstaande voorbeeld worden de namen van stations naar ascii uitvoer geschreven. Dit voorbeeld is opgenomen omdat het schrijven van de waarden van string variabelen naar ascii uitvoer niet goed gaat in box-formaat en matlab-formaat. Het wegschrijven van de strings naar gewoon formaat gaat echter wel goed, zonder dat de dimensie of de lengte van de variabele aangepast hoeft te worden.

Het wegschrijven van strings in box- of matlab-formaat kan wel worden nagebootst. In het onderstaande voorbeeld wordt de uitvoer volgens gewoon formaat weggeschreven waarbij het matlab-formaat met behulp van de **PRINT** opdracht letterlijk ingetikt wordt.

```
# =====
#   uitvoer ascii   = stations.asc (gewoon formaat)
# =====
name='namwl'
outfile='stations.asc'
nobox
go

# =====
#   uitvoer ascii   = stations.m (matlab formaat)
# =====
outfile='stations.m'
print 'namwl=[... '
format='\''\20($)\''
go
print '];'
```

6.8 Voorbeeld: tijden.inp

In het onderstaande voorbeeld wordt tijdreeksuitvoer weggeschreven naar een binaire matlab uitvoer file. Ook wordt laten zien hoe tijdreeksuitvoer weggeschreven kan worden in ascii matlab-formaat. Indien per tijdstip een matrix moet worden weggeschreven met een andere naam (bijvoorbeeld de tijd in de naam), wordt aangeraden dit met behulp van een run-script te doen, zie paragraaf 6.10.

In het volgende voorbeeld wordt tijdreeksuitvoer van de variabele "sep" (waterstand in rekenpunten) voor de eerste 15 tijdstippen weggeschreven naar een binaire matlab uitvoer file.

Het wegschrijven van tijdreeksuitvoer in ascii matlab files is niet standaard mogelijk. Dit kan echter wel nagebootst worden door de uitvoer volgens het gewone format weg te schrijven waarbij het matlab-formaat letterlijk met behulp van de **PRINT** opdracht ingetikt wordt.

Opmerking Dit is echter alleen mogelijk indien gegevens voor kleine deelgebieden weggeschreven worden. In het onderstaande voorbeeld worden namelijk alle gegevens per tijdstip in één rij weggeschreven. MATLAB kan slechts een bepaalde hoeveelheid gegevens per rij aan.

```
# =====
#   uitvoer binair   = waterlv.mat
# =====
lgrid
matfile='waterlv.mat'
mattime

#   variabele SEP = waterstand
name='sep'
matname='waterstand'
time=-1 4
# time=-1, omdat dan bij het allereerste
#   tijdstip op SDS file wordt begonnen
repeat=15
go

# =====
#   uitvoer ascii   = waterlv.m
# =====
outfile='waterlv.m'

#   variabele SEP = waterstand
name='sep'
mnmn 50,50,55,55
nobox
print 'sep=[ ...'
time=-9 4
repeat=15
format=':\8($$$$.$$$ ) ...'
# de dubbele punt in het format zorgt ervoor dat
# na het laatste veld geen ... worden geschreven,
# d.w.z. de uitvoer wordt afgekapt als er geen
# waarden meer te printen zijn

pritim='$$$$$ ...'
```

```
go
print '];'
```

6.9 Run-script: **rsds.gain**

Met behulp van het onderstaande run-script wordt de Kalman gain per station weggeschreven naar een binaire matlab file.

Het script wordt uitgevoerd door het intikken van **rsds.gain** gevolgd door een tweetal argumenten:

rsds.gain <arg1> <arg2>

waarbij:

arg1	=	run-identificatie (opc)	
arg2	=	soort gain:	standaard = sp (waterstand)
		anders=	u (u-snelheid water)
			v (v-snelheid water)
			wu (u-snelheid wind)
			wv (v-snelheid wind)

Argument 2 geeft de soort gain aan. Indien de gain voor de waterstand wordt gekozen, kan het tweede argument weggelaten worden, aangezien dit standaard is. Als een andere gain wordt gekozen, dan moet dit wel als tweede argument worden meegegeven. Bijvoorbeeld:

rsds.gain waq wu (de gain voor de u-snelheid wind)

Met behulp van het run-script wordt vervolgens de binaire matlab file "gain<soort>.mat" aangemaakt.

Daarna wordt de gain per station gelezen en weggeschreven, waarbij per station een andere matlab naam voor de variabele "gain<soort>" wordt meegegeven.

```
error=0
if [ -z "$1" ]; then
    echo "Error: no runid given (first parameter)"
    error=1
fi
if [ -z "$2" ]; then
    gain='sp'
else
    gain=$2
fi
if [ $error = 1 ]; then
    exit
fi
cat -s <<END >/tmp/rsds$$
    name='/kalman/int/nowl' go
    exit
END
rsds.pl -runid $1 -infile /tmp/rsds$$ -outfile
/tmp/rsds1$$ -back n >rsds-m.$1 2>&1
read nowl </tmp/rsds1$$
echo "nowl=$nowl"

cat -s <<END >/tmp/rsds$$
```

```
        matfile='gain$gain.mat'
        lgrid
#
# Waterlevel station 1:
    name    = 'gain$gain'
    matname= 'gain$gain_1'
    go
END
iwl=1
while [ "$iwl" -lt "$nowl" ]; do
    let iwl=iwl+1
    cat -s <<END >>/tmp/rsds$$
#
# Waterlevel station $iwl:
    matname= 'gain$gain_$iwl'
    start   = '+mnmaxk'
    go
END
done
rsds.pl -runid $1 -infile /tmp/rsds$$ -outfile
/tmp/rsds1$$ -back n >rsds-m.$1 2>&1
rm -f /tmp/rsds$$
rm -f /tmp/rsds1$$
```

6.10 Run-script: rsds.tijd

Met behulp van het onderstaande run-script kunnen de waarden van een variabele op alle tijdstippen in de SDS file worden weggeschreven naar binaire matlab files (per tijdstip één file).

Het script wordt uitgevoerd door het intikken van **rsds.tijd** gevolgd door een tweetal argumenten:

rsds.tijd <arg1> <arg2>

waarbij:

arg1 = run-identificatie (opc)
arg2 = naam van variabele

Als argument 2 wordt de naam van de variabele opgegeven, bijvoorbeeld:

rsds.tijd waq sep (de waterstand in rekenpunten)

In het onderstaande script is uitgegaan van variabelen met een lgrid-index. Als een variabele met een andere index wordt opgegeven, dan moet het script worden aangepast.

Eerst wordt door het run-script opgevraagd: de eerste tijdstap, de laatste tijdstap en het tijdsinterval op de SDS file. Deze gegevens verschijnen ook in de standaard uitvoer (scherm of uitvoer file).

Vervolgens wordt voor iedere tijdstap een binaire matlab file aangemaakt met de naam "rsds_<tijdstip>.mat" en worden de waarden van de variabele op dat tijdstip in de matfile geschreven.

```
error=0
if [ -z "$1" ]; then
    echo "Error: no runid given (first parameter)"
    error=1
else
    runid=$1
    sdsfile=SDS-$runid
    if [ ! -f $sdsfile ]; then
        echo
        echo "***Error: $sdsfile does not exist!"
        echo
        error=1
    fi
fi

if [ -z "$2" ]; then
    echo "Error: no name of variable given (third
parameter)"
else
    var=$2
fi
if [ $error = 1 ]; then
    exit
fi
cat -s <<END >/tmp/rsds$$
```


7 VOORBEELDEN SDS DOCUMENTATIE FILES

In dit hoofdstuk zijn de volgende SDS documentatie files opgenomen:

- SDS documentatie file voor WAQUA (Hoofdstuk 7.1)
- SDS documentatie file voor WAQAD (Hoofdstuk 7.2)
- SDS documentatie file voor KALMAN (Hoofdstuk 7.3)
- SDS documentatie file voor meetwaarden (Hoofdstuk 7.4)
- SDS documentatie file voor windgegevens (Hoofdstuk 7.5)

7.1 SDS documentatie file voor WAQUA

De SDS documentatie file voor WAQUA (sds_WAQUA.doc) bevindt zich op het rekencluster "compcl" op de directory:

```
/users/applic/waqad/bin
```

Hieronder is deze documentatie file opgenomen. Voor een beschrijving van de variabelen en paden in deze file wordt verwezen naar "Local data Structure WAQUA-in-SIMONA", [2].

```
#
# -----
# WAQUA problem field FLOW
# -----
#
1  'MESH'  LOAD
#
2      'IDIMEN'      type='int'    length=1
          'ndim'      'mmax'      'nmax'      'mnmax'      'mnmaxk'    'nenclo'
          'ldam'      'nocols'    'noroco'    'norows'    'nslu'      'nsluv'
          'nslv'      'nto'      'iadlnd'    'kurflg'    'nrou'      'kmax'
          'kmaxs'
#
2      'CONMSH'      type='real'   length=1
          'al'        'dy'        'dksi'      'anglat'    'rlambd'    'fi'
          'grdang'    'dlambd'    'dfi'      'rearth'
#
length=3  length='*/mesh/idimen/nenclo'  type='int'
2      'IENCLO'
#
2      'IOPEN'      type='int'
          length='*/mesh/idimen/nto'
          'kb1m'      'kb1n'      'kb2m'      'kb2n'
#
2      'BARPOS'      type='int'
          length='*/mesh/idimen/nsluv'
          'mbar'      'nbar'      'ibuv'
#
2      'IROGEO'      type='int'
          length=3  length='*/mesh/idimen/norows'
          'irows'
          length=3  length='*/mesh/idimen/nocols'
          'icols'
```

```

#      length='/mesh/idimen/nmax'   length='*/mesh/idimen/mmax'   type='int'
2      'LGRID'
#
2      'IDAM'           type='int'
      length='/mesh/idimen/ldam'
      'mdam'         'ndam'
#
      length='/mesh/idimen/mnmaxk'   type='real'
2      'H'
#
2      'IVARWP'       type='int'
      length=1
      'ipm'          'jpm'          'mpf'          'npf'          'iwm'          'jwm'
      'mwf'          'nwf'
#
2      'CURVIL'       type='real'
      length='/mesh/idimen/mnmaxk'
      'guu'          'gvv'          'xdep'          'ydep'          'xzeta'          'yzeta'
#
2      'NAMMSH'       type='char*80'
      length='/mesh/idimen/nsluv'
      'nambar'
      length='/mesh/idimen/nto'
      'namt1'        'namt2'
#
2      'WEIPOS'       type='int'
      length='/mesh/idimen/nrou'
      'mwei'         'nwei'         'iweiu'        'iweiv'
#
      length='/mesh/idimen/kmaxs'   type='real'
2      'HLAY'
#
      length='/mesh/idimen/kmaxs'   type='real'
2      'INDLAY'
#
1      'MESH01_GENERAL_COOR'
1      'MESH01_GENERAL_LGRID'
1      'MESH01_GENERAL_MESHNAMEs'
1      'MESH01_GENERAL_DIMENSIONS'
#
#
#
1      'SOLUTION_DRYWET' type='int'
      length='/mesh/idimen/mnmaxk'
      'khu'          'khv'
#
#
#
1      'LAYER_INTERFACES' type='int'
      length=1
      length='+/mesh/idimen/kmax'
      length='*/mesh/idimen/mnmaxk'
      'zk'
#
#
#
1      'CONTROL_PROCES' type='int'
      length=1
      'iflow'        'itrans'        'ipart'         'imodel'        'istres'        'iusrfl'
      'iusrtr'        'iusrpa'        'ikalnmn'
#
#
#
1      'COEFF_GENERAL'
#

```

```

2      'ICGENA'      type='int'
      length=1
      'notgwn'  'isvwp'  'icdflg'  'iwlopf'
#
2      'RCGENB'      type='real'
      length=1
      'ag'      'dair'  'dwat'  'wconv'  'wstr'  'wscdv1'
      'wscdv2'  'cdv1'  'cdv2'
#
2      'WIND'      type='real'
      length=1
      'pwindi'  'pwinai'
      length='/coeff_general/icgena/notgwn'
      'timdis'  'pwind'  'pwinda'
#
2      'IREFSW'      type='char*80'
      length=1
      'wpnsds'  'wpnexp'
#
      length='/mesh/idimen/mnmaxk'  length='*/mesh/idimen/kmax'  type='real'
2      'DIFCO'
#
#
#
1  'PROBLEM_FLOW'  LOAD
#
#
1  'CONTROL_FLOW'
#
2      'ICONTA'      type='int'
      length=1
      'nopow'  'nsrc'  'ntof'  'ntot'  'kc'  'nowl'
      'nocur'  'ntra'  'ntrav'  'notbar'  'nottid'  'notdis'
#
2      'ICONTB'      type='int'
      length=1
      'idryfl'  'iter1'  'iter2'  ''  ''  'ldengr'
      length=18
      ''
      length=1
      'nprc'  ''  'nprsep'  'nprvcu'  'nprvcv'  'nprvcw'
      ''  'nprvml'  'nprvmu'  'nprvmv'  'roumet'  ''
      ''  ''  ''  ''  'iturfl'  'irwall'
      'logbou'  'lrich'
#
2      'RCONTA'      type='real'
      length=1
      'dtmin'  ''  'tstart'  'tstop'  ''  'eps'
      ''  'ticval'  ''  ''  'var'  ''
      ''  ''  ''  ''  ''  ''
      ''  ''  ''  ''  'tlfsmo'  ''
      ''  ''  ''  ''  ''  ''
      'salw'  ''  ''  'rho'  'alph0'  'tempw'
      'thetac'  'rfelag'  'rfelng'  'hkrdum'  ''  'vico'
      ''  ''  'defviv'  'z0'  'teta'  'cmukl'
#
2      'RCONTB'      type='real'
      length=42
      'tprint'
      length=1
      'tfhisp'  'tihisp'  'tlhisp'  'tfstap'  'tistap'  'tlstap'
      length=52
      ''
      length=1

```

```

                'tfint'   'tiint'   'tlint'   'tfmapf' 'timapf' 'tlmapf'
                'tfhist' 'tihist' 'tlhist' 'tfrst'  'tirst'  'tlrst'
#
#
#
1  'PROBLEM_FLOW'
#
2  'IROBOU'      type='int'
    length=2    length='*/mesh/idimen/norows'
    'irowb'
    length=2    length='*/mesh/idimen/nocols'
    'icolb'
#
2  'IDISCH'      type='int'
    length='/control_flow/icona/nsrc'
    'mdis'     'ndis'     'kdis'
#
2  'NAMPRB'      type='char*80'
    length=1
    'itdate'
    length='/control_flow/icona/nsrc'
    'namdis'
#
1  'PROBLEM_FLOW01_GENERAL_IFUNC'
1  'PROBLEM_FLOW01_GENERAL_GLOBAL'
1  'PROBLEM_FLOW01_GENERAL_FREEDOM'
1  'PROBLEM_FLOW01_GENERAL_SOLUTIONTYPES'
#
#
#
1  'FORCINGS_FLOW'
#
    length='/mesh/idimen/nto'      type='int'
2  'KBO'
#
    length='/mesh/idimen/nto'      type='real'
2  'ALFA'
#
2  'TIDINT'      type='real'
    length='/control_flow/icona/ntot'
    'tidnta'   'tidntb'
#
2  'TIMSER'      type='real'
    length='/control_flow/icona/nottid'
    'timtid'
    length='*/control_flow/icona/ntot'
    'tid1'     'tid2'
#
2  'TIDINF'      type='real'
    length='/control_flow/icona/ntof'
    'tidnfa'   'tidnfb'
#
2  'AZERO'      type='real'
    length='/control_flow/icona/ntof'
    'azeroa'   'azerob'
#
2  'AMPL'      type='real'
    length='/control_flow/icona/ntof'
    length='*/control_flow/icona/kc'
    'ampla'    'amplb'
#
    length='/control_flow/icona/kc'      type='real'
2  'OMEGA'
#
2  'PHASE'      type='real'

```

```

length='/control_flow/icona/ntof'
length='*/control_flow/icona/kc'
  'phasea'  'phaseb'
#
2  'DISCHT'      type='real'
   length='/control_flow/icona/nsrc'
   'dischi'
   length='/control_flow/icona/notdis'
   'timdis'
   length='*/control_flow/icona/nsrc'
   'disch'
#
#
#
1  'COEFF_FLOW'
#
length=6  length='*/mesh/idimen/nsluv'  type='real'
2  'BARMU'
#
2  'BARDIM'      type='real'
   length='/mesh/idimen/nsluv'
   'brat'  'gate'  'sill'
   length='/control_flow/icona/notbar'
   'timdis'
   length='*/mesh/idimen/nsluv'
   'brattm' 'gatetm' 'silltm'
#
2  'CMAN'        type='real'
   length='/mesh/idimen/mnmaxk'
   'cmanur'  'cmanvr'
#
2  'WEIDIM'      type='real'
   length='/mesh/idimen/nrou'
   'hkru'
   length=2  length='*/mesh/idimen/nrou'
   'crouu'
   length='/mesh/idimen/nrou'
   'hkrv'
   length=2  length='*/mesh/idimen/nrou'
   'crouv'
#
2  'GROTYP'      type='real'
   length='/mesh/idimen/nrou'
   'groynu'  'groynv'
#
   length=1
   length='*/mesh/idimen/kmax'
   length='*/mesh/idimen/mnmaxk'
2  'VICOWG'      type='real'
#
#
#
1  'SOLUTION_FLOW'
length='/mesh/idimen/mnmaxk'  type='real'
2  'SEP'
length='*/mesh/idimen/kmax'  type='real'
2  'UP'
2  'VP'
2  'W'
   'w1'  length='/mesh/idimen/mnmaxk'  'w2'
length='/mesh/idimen/mnmaxk'  length='*/mesh/idimen/kmax'  type='real'
2  'WPHYS'
length='/mesh/idimen/mnmaxk'  type='real'
2  'CZ'
   'czu'  'czv'
#

```

```

#
#
1  'INTEGRAL_FLOW'
#
2      'TIMINT'          type='real'
        length=1
        'tfintg'  'tlintg'  'tintg'
#
        length='/mesh/idimen/mnmaxk'  type='real'
2      'SEPNOW'
2      'DEPINT'
2      'UPINT'
2      'VPINT'
2      'DISUNT'
2      'DISVNT'
#
#
#
1  'RESTART_FLOW'    type='real'
#
2      'EXTRES'        type='real'
        length=1
        'udir'        'vdir'
        length='/mesh/idimen/nsluv'
        'cbuv'
#
        length='/mesh/idimen/mnmaxk'  length='*/mesh/idimen/kmax'  type='real'
2      'VICOW'
#
#
#
1  'TIMEHISTORIES_FLOW'
#
2      'TIMHIS'        type='real'
        length='/control_flow/icona/nowl'
        'zwl'
        length='/control_flow/icona/nocur'
        length='*/mesh/idimen/kmax'
        'zcur'        'zcuru'        'zcurv'
        length='/control_flow/icona/ntra'
        'ctr'        'fltr'
        length='/control_flow/icona/ntrav'
        'ctrv'        'fltrv'
        length='/mesh/idimen/nsluv'
        'barq'        'basepa'        'basepb'        'bavela'        'bavelb'
        length='/control_flow/icona/nocur'
        'zcurw'
        length='*/mesh/idimen/kmax'
        'zcurw1'        'zcurwp'
        length='/control_flow/icona/nocur'
        'zkcur'
        length='*/mesh/idimen/kmax'
        'zkcurl'
#
        length='/mesh/idimen/nsluv'  type='int'
2      'KCOND'
#
#
#
1  'CHECKPOINTS_FLOW'
#
2      'IWLPT'        type='int'
        length='/control_flow/icona/nowl'
        'mw1'        'nwl'
#

```

```

2      'ICURPT'          type='int'
        length='/control_flow/icona/nocur'
        'mc'          'nc'
#
2      'ICROSU'          type='int'
        length='/control_flow/icona/ntra'
        'mit'         'nit1'      'nit2'
#
2      'ICROSV'          type='int'
        length='/control_flow/icona/ntrav'
        'nit'         'mit1'      'mit2'
#
2      'NAMCHK'          type='char*80'
        length='/control_flow/icona/nowl'
        'namwl'
        length='/control_flow/icona/nocur'
        'namc'
        length='/control_flow/icona/ntra'
        'namtra'
        length='/control_flow/icona/ntrav'
        'namtrv'
#
#
#
1      'DISPLAYS'
#
2      'IDISPL'          type='int'
        length=1
        'iwldep'      'isocol'    'isonum'    'nctitl'
#
2      'RDISPL'          type='real'
        length=1
        'arrlen'      'clkrad'    'dxdpy'     'fmtiso'    'siziso'    'tang'
        'vectw'       'xleft'     'xright'    'ybot'      'ytop'      ''
        ''            ''          ''          'xdelt'     'ydelt'     ''
        length=4
        'hx'          'hy'        'hz'
#
2      'CEPLOT'          type='char*80'
        length=1
        ''            ''          'pwunit'    ''
#
2      'ILAND'           type='int'
        length=1
        'ncnst'       'nsub'      'nolan'     'nolin'     'nopt'      'ista'
        'leng'
        length='/displays/iland/leng'
        'land'
#
2      'RLIN'            type='real'
        length='/displays/iland/nolin'
        'blin'        'bmod'      'bspc'
#
2      'CORLAN'          type='real'
        length='/displays/iland/nopt'
        'xland'       'yland'
#
2      'IPERM'           type='int'
        length=1
        'ncnst'       'nsub'      'notip'     'istal'     'leng1'     'ista2'
        'leng2'
        length='/displays/ipermleng1'
        'iticv'
        length='/displays/ipermleng2'

```

```

        'itich'
#
2      'PERMNT'      type='real'
        length='/displays/iper/perm/notip'
        'titlm'    'titln'    'titlw'    'titor'    'titsz'
#
        length='/displays/iper/perm/notip'    type='char*80'
2      'TTITL'
#
#
#
1      'RUNIDENT'    type='char*80'
        length=1
        'htitl'    'modid'    'runid'    'titl'    ''    'verlds'
        'verpre'    ''    ''    'datpre'
#
# -----
# WAQUA problem field TRANS
# -----
#
1      'PROBLEM_TRANS'
#
#
#
1      'CONTROL_TRANS'
#
2      'ICONTA'      type='int'
        length=1
        'nopow'    'nsrc'    'nopol'    'ntra'    'ntrav'    'ntopt'
        'nottid'    'notdis'    'notcdi'    'notqnr'    'notdry'    'notrhu'
        'nottwm'    'notqsc'
#
2      'ICONTB'      type='int'
        length=1
        'ktemp'    'lmax'    'lsal'    'ltemp'    'lerg'    'leps'
        'ldifco'
#
2      'ICONTC'      type='int'
        length='/control_trans/icontb/lmax'
        'nprrr'    'kpflag'
#
2      'RCONTA'      type='real'
        length=1
        'dtmin'    'tstart'    'tstop'    ''    ''    ''
        ''    ''    ''    ''    'fclou'    ''
        'sarea'    'chesal'    ''    ''    ''    ''
        ''    ''    'depdiv'    'tetatr'    'alfmu1'    'alfmu2'
        'betmul'    'betmu2'    ''    ''    ''    ''
#
2      'RCONTB'      type='real'
        length=42
        'tprint'
        length=1
        'tfhisp'    'tihisp'    'tlhisp'    'tfstap'    'tistap'    'tlstap'
        length=12
        ''
        length=1
        'tfmapf'    'timapf'    'tlmapf'    'tfhist'    'tihist'    'tlhist'
        'tfrst'    'tirst'    'tlrst'
#
#
#
1      'PROBLEM_TRANS'
#
        length=2    length='*/mesh/idimen/noroco'    type='int'
2      'IROBOU'

```



```

#
2   'IDISCH'          type='int'
    length='/control_trans/icona/nsrc'
    'mdis'          'ndis'          'kdis'
#
2   'NAMPRB'         type='char*80'
    length=1
    'itdate'
    length='/control_trans/icona/nsrc'
    'namdis'
    length='/control_trans/iconb/lmax'
    'polt'          'poltun'
#
#
#
1   'FORCINGS_TRANS'
#
    length='/mesh/idimen/nto'      type='int'
2   'KB0'
#
2   'TCRET'          type='real'
    length='/mesh/idimen/nto'
    'tcreta'        'tcretb'
#
2   'TIMSER'         type='real'
    length='/mesh/idimen/nto'
    length='*/control_trans/iconb/lmax'
    'trbnai'        'trbnbi'
    length='/control_trans/icona/nottid'
    'timtid'
    length='*/mesh/idimen/nto'
    length='*/control_trans/iconb/lmax'
    'trbnda'        'trbndb'
#
2   'DISCHT'         type='real'
    length='/control_trans/icona/nsrc'
    'dischi'
    length='/control_trans/icona/notdis'
    'timdis'
    length='*/control_trans/icona/nsrc'
    'disch'
#
2   'CONTIM'         type='real'
    length='/control_trans/icona/nsrc'
    length='*/control_trans/iconb/lmax'
    'sinti'
    length='/control_trans/icona/notcdi'
    'timdis'
    length='*/control_trans/icona/nsrc'
    length='*/control_trans/iconb/lmax'
    'sint'
#
2   'QNRFLT'         type='real'
    length=1
    'qnfrli'
    length='/control_trans/icona/notqnr'
    'timdis'        'qnfrl'
#
2   'TDRYBT'         type='real'
    length=1
    'tdrybi'
    length='/control_trans/icona/notdry'
    'timdis'        'tdryb'
#

```

```

2      'RHUMIT'          type='real'
      length=1
      'rhumii'
      length='/control_trans/icona/notrhu'
      'timdis' 'rhumid'
#
2      'TWMST'          type='real'
      length=1
      'twmsi'
      length='/control_trans/icona/nottwm'
      'timdis' 'twms'
#
2      'QSCT'          type='real'
      length=1
      'qsci'
      length='/control_trans/icona/notqsc'
      'timdis' 'qsc'
#
#
#
1      'COEFF_TRANS'
#
#
#
1      'SOLUTION_TRANS' type='real'
#
      length='/control_trans/iconb/lmax'
      length='*/mesh/idimen/kmax'
      length='*/mesh/idimen/mnmaxk'
2      'RP'
#
#
#
1      'RESTART_TRANS'
#
      length='/control_trans/icona/ntopt'      type='integer'
      length='*/mesh/idimen/kmax'
2      'NFLI'
#
      length='/control_trans/icona/ntopt'      type='real'
      length='*/mesh/idimen/kmax'
      length='*/control_trans/iconb/lmax'
2      'DFL'
#
      length='/mesh/idimen/kmax'              type='real'
      length='*/mesh/idimen/mnmaxk'
2      'DIFCW'
#
#
#
1      'TIMEHISTORIES_TRANS'
      length='/mesh/idimen/kmax'
      length='*/control_trans/icona/nopol'
      length='*/control_trans/iconb/lmax'
      'gro'
      length='/control_trans/icona/ntra'
      length='*/control_trans/iconb/lmax'
      'adtr' 'atr' 'dtr'
      length='/control_trans/icona/ntrav'
      length='*/control_trans/iconb/lmax'
      'adtrv' 'atr'v' 'dtrv'
      length='/mesh/idimen/kmax'
      length='*/control_trans/icona/nopol'
      'zkpol'
      length='/control_trans/icona/nopol'

```

```

                'zkpoll'
#
#
#
1  'CHECKPOINTS_TRANS'
#
2    'IPOLPT'          type='int'
        length='/control_trans/icona/nopol'
        'mpol'        'npol'
#
2    'ICROSU'          type='int'
        length='/control_trans/icona/ntra'
        'mit'         'nit1'    'nit2'
#
2    'ICROSV'          type='int'
        length='/control_trans/icona/ntrav'
        'nit'         'mit1'    'mit2'
#
2    'NAMCHK'          type='char*80'
        length='/control_trans/icona/nopol'
        'nampol'
        length='/control_trans/icona/ntra'
        'namtra'
        length='/control_trans/icona/ntrav'
        'namtrv'
#
#
#
1  'USERDATA_TRANS'
#
2    'USRDIM'          type='int'    length=1
        'lenusr'      'lenius'    'lenwrk'  'leninp'  'lensol'  'nottim'
        'nval'        'nfour'   'lenfou'  'lentdf'  'notfld'  'ntype'
#
length='/userdata_trans/usrdim/lenusr'    type='real'
2    'USER'
#
length='/userdata_trans/usrdim/lenius'    type='integer'
2    'IUSER'
#
length=2  length='*/userdata_trans/usrdim/lenfou'    type='integer'
2    'IFLAG'
#
2    'TIMSER'          type='real'
        length='/userdata_trans/usrdim/nottim'
        'timtid'
        length='*/userdata_trans/usrdim/nval'
        'timval'
#
length='/userdata_trans/usrdim/leninp'    type='integer'
length='*/mesh/idimen/mnmaxk'
2    'SPAINP'
#
length='/userdata_trans/usrdim/notfld'    type='real'
2    'TIMSPD'
#
2    'NAMUSR'
        length='/userdata_trans/usrdim/lensol'    type='char*80'
        'usrnam'  'unitus'
#
length='/userdata_trans/usrdim/lenfou'    type='real'
2    'AZERO'
#
length='/userdata_trans/usrdim/nfour'    type='real'
length='*/userdata_trans/usrdim/lenfou'

```

```

2      'AMPL'
#
length='/userdata_trans/usrdim/nfour'   type='real'
2      'OMEGA'
#
length='/userdata_trans/usrdim/nfour'   type='real'
length='*/userdata_trans/usrdim/lenfou'
2      'PHASE'
#
#
#
1      'USER_SPATIM_TRANS'
      length='/userdata_trans/usrdim/lentdf'   type='real'
      length='*/mesh/idimen/mnmaxk'
      'spatim'
#
#
#
1      'SOLUTION_USER_TRANS'
      length='/userdata_trans/usrdim/lensol'   type='real'
      length='*/mesh/idimen/mnmaxk'
      'solusr'
#
#
#
1      'TIMEHISTORIES_USER_TRANS'
      length='/userdata_trans/usrdim/lensol'   type='real'
      length='*/control_trans/icona/nopol'
      'tmhusr'
#
# -----
# WAQUA problem field STRESS-2D
# -----
#
1      'STRESS'
#
2      'S2INT'           type='int'   length=1
      'nfetch'  'nlayer'  'nfrac'   'irese'
#
2      'S2REAL'         type='real'  length=1
      'minws'   'mindep'  'npartk'
#
length='/stress/s2int/nfrac'   type='real'
2      'CONFRA'
#
length=2  length='*/stress/s2int/nlayer'   type='real'
2      'BOEXTR'
#
length='/stress/s2int/nlayer'   type='real'
2      'BOPORO'
#
2      'BOCONR'
#
2      'BOEROR'
#
2      'BOBIOR'
#
type='char*80'
2      'BONAME'
#
length=2  length='*/stress/s2int/nfetch'   type='real'
2      'WVANGL'
#
length='/mesh/idimen/mnmaxk/'  length='*/stress/s2int/nfetch'   type='real'
2      'FETCH'

```

```

#
# -----
# WAQUA problem field INPUT
# -----
#
1   'INPUT'
#
# -----
# Problem field MISC
# -----
#
1   'MISC'
#
2   'IBOUND'          type='int'
      length=2 length='*/mesh/idimen/nto'
      'ipntkb'
      length=4 length='*/mesh/idimen/nto'
      'irckb'
#
# -----
# Problem field KALMAN
# -----
#
1   'CONTROL_KALM'
#
2   'IDIMEN'          type='int'
      length=1
      'nowlk'  'nopnt'  'nmode'  'npar'  'tikal'  'usezer'
      'noistr' 'nbcirc' 'nwcirc'
#
2   'STATIONS'       type='int'
      length='*/control_kalm/idimen/nowlk'
      'lgridk'  'iobsk'
#
2   'IREFKM'         type='char*80'
      length=1
      'kalsds'  'kalexp'
#
2   'BOUND_POINTS'  type='int'
      length=1
      'nbnd'
      length=2 length='*/control_kalm/bound_points/nbnd'
      'mnbnd'
#
2   'WIND_POINTS'   type='int'
      length=1
      'nwnd'
      length=2 length='*/control_kalm/wind_points/nwnd'
      'mnwnd'
#
1   'PROBLEM_KALM'
#
2   'BOUND_WEIGHT'  type='real'
      length=2 length='*/mesh/idimen/noroco'
      length='*/control_kalm/bound_points/nbnd'
      'wghtb'
#
2   'WIND_WEIGHT'   type='real'
      length=4 length='*/mesh/idimen/mnmaxk'
      'wghtwu'  'wghtwv'
#
2   'WIND_PARPT'    type='int'
      length=4 length='*/mesh/idimen/mnmaxk'
      'ipparw'
#
1   'COEFF_KALM'

```

```
#
2      'TIME_CORR'          type='real'
      length=1
      'cortib'  'cortiw'

#
      type='real'  length='/control_kalm/idimen/nowlk'
2      'VAROBS'
#
1      'SOLUTION_KALM'
#
2      'STAT'              type='real'
      length='/control_kalm/idimen/npar'
      length='*/control_kalm/idimen/npar'
      'Q'
      length='/control_kalm/idimen/nowlk'
      length='*/control_kalm/idimen/nowlk'
      'R'

#
1      'RESTART_KALM'
#
2      'STEADY_STATE'      type='real'
      length='/mesh/idimen/mnmaxk'
      'xup'      'xvp'      'xsep'
      length=2  length='*/mesh/idimen/noroco'
      'cirkal'
      length='/mesh/idimen/mnmaxk'
      'wukal'  'wvkal'

#
2      'RRSQRT'           type='real'
      length=3
      length='*/control_kalm/idimen/nopnt'
      'xkal'
      length='/control_kalm/idimen/npar'
      length='+/control_kalm/idimen/nmode'
      length='*/control_kalm/idimen/nopnt'
      'L'

#
2      'IRRSQRT'          type='int'
      length=1
      'kalcnt'

#
1      'HISTORY_KALM'
#
2      'BOUND'            type='real'
      length=2
      length='*/mesh/idimen/nto'
      length='*/control_kalm/idimen/nbcirc'
      'cirkal'
      length=2
      length='*/mesh/idimen/nto'
      'cirkal_rest'
```

7.2 SDS documentatie file voor WAQAD

De SDS documentatie file voor WAQAD (sds_WAQAD.doc) bevindt zich op het rekencluster "compcl" op de directory:

```
/users/applic/waqad/bin
```

Hieronder is deze documentatie file opgenomen. Voor een beschrijving van de variabelen en paden in deze file wordt verwezen naar "User's guide WAQAD", [3], en "System documentation WAQAD", [4].

```
#
# -----
# ADPRO problem field ADJOIN
# -----
#
1  'ADJOIN'
#
2    'IINPUT'      type='int'
      length=1
        'maxite'  'iterf'   'iterl'   'iter'    'mxforw'  'mxback'
        'miter'   'iofcd'  'iofdp'  'iofcz'   'iofkb'   'iofvv'
        'nparam'  'ncd'    'ndp'    'ndbp'    'ncz'     'nbcz'
        'nkc'     'nbl'    'ntokb'  'mbound'  'nparkb'  'nkcpa'
        'nvv'     'nbvv'   'nsimvv' 'kmax'    'nfindf'  'ifindf'
        'iperdp'  'ipercz' 'obsmean' 'ipenal'  'iadv'    'iwgrad'
        'nsimcd'  'nsimdp' 'nsimcz' 'nsimkb'  'nsimad'  'nsim2d'
        'nowl'    'ntsat'  'nsim3d' 'l2d'     'l3d'     'version'
        'ipervv'  ''       ''       ''       ''       ''
        ''       ''       ''       ''       ''       ''
        ''       ''       ''       ''       ''       ''
        ''       ''       ''       ''       ''       ''
        length='/adjoin/iinput/ncd'
        'cdhold'      length=1  ''
        length=5
        length='*/adjoin/iinput/nbdp'
        'idarea'      length=1  ''
        length=5
        length='*/adjoin/iinput/nbcz'
        'icarea'      length=1  ''
        length=5
        length='*/adjoin/iinput/nbv'
        'ivarea'      length=1  ''
        length='/adjoin/iinput/nkc'
        'kbhc'        'kbcomb'
        length=4
        length='*/adjoin/iinput/nbl'
        'kbinbl'
        length='/adjoin/iinput/ntokb'
        'kbchp'        'kblist'  length=1  'kbl1'
        length=2
        length='*/adjoin/iinput/ntokb'
        'kbchn'
        length=2
        length='*/adjoin/iinput/mbound'
        'kbpar'        'kbmnc'

        length=2
        length='*/adjoin/iinput/ntokb'
        'kbpnts'      length=1  ''
        length=1
```

```

        'ntdp'      'ntcz'      'npoint'  'ntrian'
#   iparndp|cz bevat het puntnummer dat bij een parameter hoort
length='/adjoin/iinput/npoint'
        'iparndp'
length='/adjoin/iinput/npoint'
        'iparncz'
#   ipardp|cs bevat de (max 3) parameternummers per gridpunt
length=3
length='*/adjoin/iinput/nsimdp'
        'ipardp'
length=3
length='*/adjoin/iinput/nsimcz'
        'iparcz'
#   itriddp|cz bevat de nummers (index) van de gebruikte driehoeken
length='/adjoin/iinput/ntdp'
        'itriddp'
length='/adjoin/iinput/ntcz'
        'itricz'
#   ipoint bevat alle punten (m n p: coördinaten en "parameter doet
#   mee" indicatie: 0=ja, 1=nee)
length=3
length='*/adjoin/iinput/npoint'
        'ipoint'
#   itrrian bevat alle driehoeken
length=3
length='*/adjoin/iinput/ntrian'
        'itrrian'
length=1
        'ntvv'
#   iparnvv bevat het puntnummer dat bij een parameter hoort
length='/adjoin/iinput/npoint'
        'iparnvv'
#   iparvv bevat de (max 3) parameternummers per gridpunt
length=3
length='*/adjoin/iinput/nsimvv'
        'iparvv'
#   itrivv bevat de nummers (index) van de gebruikte driehoeken
length='/adjoin/iinput/ntvv'
        'itrivv'
#
2   'RINPUT'      type='real'
        length=1
        'parcor'  'hseci'    'depmin'  ''      ''      ''
        'charcd'  'chardp'  'charcz'  'charvv' ''      ''
length='/adjoin/iinput/nparam'
        'stdpar'
#   bij driehoeken wordt stdpar per punt opgeslagen
length='/adjoin/iinput/nsimdp'
        'stddp'
length='/adjoin/iinput/nsimcz'
        'stdcz'
length='/adjoin/iinput/nsimvv'
        'stdvv'
length='/adjoin/iinput/nparam'
length='*/adjoin/iinput/nparam'
        'cov'
length='/adjoin/iinput/ncd'
        'cdscal'
length='/adjoin/iinput/nkcpar'
        'charkb'
length='/adjoin/iinput/nkc'
        'wghtam'  'wghtph'
length='/adjoin/iinput/mbound'
        'rkbalp'  'stdakb'  'stdpkb'  length=1  ''
#   wghtdp|cz bevat de (max 3) gewichten per gridpunt
length=3

```



```

length='*/adjoin/iinput/nsimdp'
  'wghtdp'
length=3
length='*/adjoin/iinput/nsimcz'
  'wghtcz'
length=3
length='*/adjoin/iinput/nsimvv'
  'wghtvv'
length='/adjoin/iinput/nparam'
  'delfin'
#
2  'CINPUT'      type='char*80'
    length=10
    'parfil'
#
#
1  'ADJOIN_HISTORY'
#
2  'ISAVE'      type='int'
    length=1
    'back'     'forw'     'niter'     'nobs'     ''     ''
#
2  'RSAVE'      type='real'
    length=1
    'step'     'parcor'  ''         ''         ''         ''
    length='/adjoin/iinput/maxite'
    'cost'     'penalty' 'genstd'   length=1  ''
    length='/adjoin/iinput/nparam'
    'del'     'dir'     'grad'    length=1  ''
    length='/adjoin/iinput/nparam'
    length='*/adjoin/iinput/nparam'
    'hsec'
    length='/adjoin/iinput/nowl'
    'mean'     'stdev'   'difmax'   'nobs'
    length='/adjoin/iinput/nparam'
    'delcum'
    length='/adjoin/iinput/maxite'
    'steps'    length=1  'stepn'
    length='/adjoin/iinput/ntsat'
    'sattim'   'satmean' 'satstd'   'satmax'  'nsat'
    length='/adjoin/iinput/nparam'
    'costfd'
    length=1  ''
#
2  'SIMINP'     type='real'
    length='/adjoin/iinput/nsimcd'
    'cd'
    length='/adjoin/iinput/nsimdp'
    'h'
    length='/adjoin/iinput/nsimcz'
    'chezu'    'chezv'
    length='/adjoin/iinput/nsimkb'
    'ampla'    'amplb'   'phasea'   'phaseb'
    length=1
    length='+/adjoin/iinput/kmax'
    length='*/adjoin/iinput/nsimvv'
    'vvisc'
#
#
1  'ADJOIN_GRAD'
#
2  'POINTS'     type='real'
    length='/adjoin/iinput/nsimdp'
    'depth'
    length='/adjoin/iinput/nsimcz'
    'chezy'

```

```
length= '/adjoin/iinput/mbound'
length= '*adjoin/iinput/nkc'
  'bounda'  'boundp'
length=1
length= '+adjoin/iinput/kmax'
length= '*adjoin/iinput/nsimvv'
  'gradvv'
#
2  'TIME_SERIE'  type='real'
    length=1
    'timgrd'
    length= '/adjoin/iinput/nparam'
    'pargrd'
#
#
1  'ADJOIN_RESTART'
#
2  'ICONTA'      type='int'
    length=1
    'irst'      'mrst'      'maxrst'  'inirst'  ''      ''
#
2  'RCONTA'      type='real'
    length=1
    'tstart'    'tfmapf'  ''      ''      ''      ''
    length= '/adjoin_restart/iconta/maxrst'
    'times'     length=1 'times1'
    length= '/adjoin/iinput/nparam'
    'gradn'
    length= '/adjoin/iinput/nsimad'
    'uk'        'vk'        'wlk'    'ukh'    'vkh'    'wlkh'
    length= '/adjoin/iinput/nsim2d'
    'wlt'       'wlth'
    length= '/adjoin/iinput/nsim3d'
    'ut'        'vt'        'uth'    'vth'
```

7.3 SDS documentatie file voor KALMAN

De SDS documentatie file voor KALMAN (sds_KALMAN.doc) bevindt zich op het rekencluster "compcl" op de directory:

```
/users/applic/waqad/bin
```

Hieronder is deze documentatie file opgenomen. Voor een beschrijving van de variabelen en paden in deze file wordt verwezen naar "User's guide KALMAN", [5], en "System documentation KALMAN", [6].

```
#
# -----
# Problem field KALMAN
# -----
#
1  'KALMAN'
#
2    'INT'          type='int'
      length=1
      'nopnt'      'nowl'      'mbound'  'nwnd'      'nbnd'
      length=4 length='*/mesh/idimen/noroco'
      'ibroco'
      type='int'   length='/kalman/int/nowl'
      'ilgrwl'
      length=1
      'last'
#
2    'REAL'        type='real'
      length=1
      'cortiw'     'cortib'
      length=1
      'last'
#
2    'GAIN'        type='real'
      length='/mesh/idimen/mnmaxk'
      length='*/kalman/int/nowl'
      'gainsp'     'gainu'     'gainv'     'gainwu'     'gainwv'
      length='/kalman/int/nowl'
      length='*/kalman/int/mbound'
      'gainb'     length='/kalman/int/nowl' 'gainb0'
      length=1
      'last'
```

7.4 SDS documentatie file voor meetwaarden

De SDS documentatie file voor de meetwaarden (sds_OBSERVED.doc) bevindt zich op het rekencluster "compcl" op de directory:

/users/applic/waqad/bin

Hieronder is deze documentatie file opgenomen. Voor een beschrijving van de variabelen en paden in deze file wordt verwezen naar de User's guides van WAQAD en KALMAN ([3] en [5]) en naar de systeem documentatie van WAQAD en KALMAN ([4] en [6]).

```
#
# -----
# Problem field OBSERVED
# -----
#
# 1 'COEFF_OBS'
#
# 2   'ICGENA'      type='int'
#         length=1
#         'nowl'    'msat'    'ntsat'
#         length='/coeff_obs/icgena/nowl'
#         'nwl'    'mwl'    'lgridp'  'nsep'
#         length='/coeff_obs/icgena/ntsat'
#         'nsat'
#
# 2   'ICGENB'      type='int'
#         length=1
#         'ivers'
#         length=2  length='*/coeff_obs/icgena/nowl'
#         'laysta'
#         length='/coeff_obs/icgena/nowl'
#         'itype'  'izref'  'itpref'
#         length=20 length='*/coeff_obs/icgena/nowl'
#         'lgrsta'
#
# 2   'RCGENA'      type='real'
#         length='/coeff_obs/icgena/nowl'
#         'varobs'
#         length='/coeff_obs/icgena/ntsat'
#         'sattim'
#
# 2   'RCGENB'      type='real'
#         length=3  length='*/coeff_obs/icgena/nowl'
#         'staxyz'
#         length=2  length='*/coeff_obs/icgena/nowl'
#         'stamm'
#         length='/coeff_obs/icgena/nowl'
#         'stadep'  'dummy'
#         length=16 length='*/coeff_obs/icgena/nowl'
#         'stawgt'
#
# 2   'CCGENA'      type='char*80'
#         length='/coeff_obs/icgena/nowl'
#         'stanam'
#
# 2   'WLEVELS'     type='real'
#         length='/coeff_obs/icgena/nowl'
#         'sep'
#
```

```
2      'WAQVAL'      type='real'
      length='/coeff_obs/icgena/nowl'
      'pred'
#
2      'TRACKS'      type='real'
      length='/coeff_obs/icgena/msat'
      'satm'      'satn'      'satobs'
#
1      'STAT_OBS'
      type='int'      length='/coeff_obs/icgena/nowl'
      'numobs'
      type='real'      length='/coeff_obs/icgena/nowl'
      'mean'      'stdev'      'maxdev'
      type='int'      length='/coeff_obs/icgena/ntsat'
      'numsat'
      type='real'      length='/coeff_obs/icgena/ntsat'
      'satmean'      'satstd'      'satmax'
      length='*/coeff_obs/icgena/msat'
      'waqsep'
```

7.5 SDS documentatie file voor wind

De SDS documentatie file voor windgegevens (sds_WIND.doc) bevindt zich op het rekencluster "compcl" op de directory:

```
/users/applic/waqad/bin
```

Hieronder is de documentatie file opgenomen. Voor een beschrijving van de variabelen in deze file wordt verwezen naar "User's guide WAQUA", [7].

```
#
# -----
# problem field WIND
# -----
#
# 'MESH' LOAD
#
# 1 'MESH01_SPECIFIC_IDIMEN' type='int' length=1
#   'mmax' 'nmax'
#   (Opmerking Dit conflicteert met de MESH in sds_WAQUA.doc)
#
# 'PROBLEM' LOAD
#
# 1 'CONTROL_SVWP' type='int'
#
# 2 'ICWINA' type='int'
#   length=1
#   'nwtim' '' '' '' '' ''
#   '' '' '' '' '' ''
#   '' '' '' '' '' ''
#   '' ''
#
#   length='/control_svwp/icwina/nwtim' type='real'
# 2 'WINTIM'
#
# 'SOLUTION_WIND'
#
#   length='/mesh01_specific_idimen/mmax'
#   length='*/mesh01_specific_idimen/nmax' type='real'
#   'windx' 'windy'
#
# 'SOLUTION_PRESS'
#
#   length='/mesh01_specific_idimen/mmax'
#   length='*/mesh01_specific_idimen/nmax' type='real'
#   'press'
#
#
# 'COEFFICIENTS'
#
#   length=1 type='real'
#   'prmean' '' '' '' '' ''
#   '' '' '' '' '' ''
#   '' '' '' '' '' ''
#   '' ''
```

```
#  
#  
1  'IDENTIFICATION'      type='char*80'  
#  
    length=1  
    'verlds'  
    length=3  
    'idtext'  
    length=1  
    'refdat'  
    length=16  
    ''
```


INDEX

BOX.....	13, 19, 24, 52, 55, 63
CONVENTIES ..	2
DEFAULT.....	14, 31
DIMEN.....	13, 20, 51, 52, 58
DUMVAL.....	13, 23, 52
ECHO.....	14, 32, 65, 67
EXIT.....	14, 32, 65, 67
FORMAT.....	13, 26, 52, 55, 63, 64, 68
GO.....	13, 23, 52, 55, 57, 58, 59, 61, 63, 64, 65, 68
LENGTH.....	13, 19, 52, 83, 87, 88, 90
LGRID.....	13, 17, 52, 55, 57, 59, 61, 64, 65, 68
MATFILE.....	14, 29, 57, 58, 59, 61, 64, 65, 67
MATLAB.....	13, 25, 51, 52, 55, 57, 58, 59, 61, 63, 64, 65, 67
MATNAME.....	14, 29, 57, 62, 64, 66
MATTIME.....	14, 30, 52, 64
MNMN.....	13, 21, 52, 55, 64
NAME.....	13, 17, 52, 55, 57, 58, 59, 61, 63, 64
NLEVEL.....	14, 32
NOBOX.....	13, 24, 63, 64
NOMATLAB.....	13, 25, 55
NOTATIES.....	2
OUTFILE.....	14, 29, 58, 59, 61, 63, 64
PRINT.....	13, 26, 56, 63, 64
PRITIM.....	13, 28, 52, 64
REOPEN.....	14, 32
REPEAT.....	13, 23, 52, 64
SHOW.....	14, 30, 52
START.....	13, 18, 52, 61, 66
TIME.....	13, 21, 52, 64, 68
TREE.....	14, 31, 53
TYPE.....	13, 22, 52, 83, 87, 88, 90