

Memo

Datum

2 november 2022

Contactpersoon

Frans Roelofsen

Doorkiesnummer

+31(0)88 335 7149

E-mail

Frans.Roelofsen@deltares.nl

Aantal pagina's

1 van 10

Onderwerp

Memo iMOD testbanken

1 Actualisatie en uitbreiding van iMOD testbanken

1.1 Kader

In 2021 en 2022 zijn belangrijke stappen gezet in de actualisatie en uitbreiding van de testbanken voor de iMOD producten. Deze actualisatie was onderdeel van het NHI project "Beheer en onderhoud software NHI".

Deltares ontwikkelt sinds 2006 aan iMOD en gerelateerde modelcodes en software. Testbanken zijn onmisbaar voor de ontwikkeling van software en daarom wordt elke iMOD tool aan tests onderworpen. Hoewel voor de gebruikers niet zichtbaar zijn testbanken voor ontwikkelaars essentieel om blijvend feed back te krijgen over aanpassingen en uitbreidingen in de software.



Dit memo geeft een overzicht van de verschillende testbanken en testmethodes die er zijn voor iMOD producten en welke uitbreidingen hebben plaatsgevonden binnen het NHI project.

1.2 Uitgangspunten iMOD Testbanken

Voor het deelproject "actualisatie en uitbreiding testbanken" zijn enkele uitgangspunten gedefinieerd:

- *De bestaande testbank iMODFLOW2005 wordt niet verder uitgebreid en aangepast.* De reden hiervoor is dat de USGS MODFLOW2005 niet meer ondersteunt waarop Deltares heeft besloten om iMODFLOW2005 te gaan uitfaseren. Alle ontwikkelingen die Deltares doet, zijn gericht op MODFLOW6 en bijbehorende pre- en postprocessing software.
- *Op 2 punten wordt de migratie van de bestaande testbank ingezet:*
 - o Migratie van iMODFLOW2005 naar MODFLOW6
 - o van een stand-alone testomgeving naar TeamCity
- Uitbreiding van de testbank met een LHM light case (iMODFLOW2005)

1.3 Samenvatting uitgevoerde acties

Het project heeft gelopen in 2021 en 2022 met een nadruk op uitvoering in het laatste jaar.

Uitgevoerd in 2021

- Inventarisatie bestaande testbanken en oriëntatie TeamCity
- Deelmodel IBRAHYM uit de iMODFLOW2005 testbank omgezet naar MODFLOW6
- Niet stationaire versie van LHM Light toegevoegd aan testbank iMODFLOW2005.
- Adviesmemo over de wijze waarop marktpartijen in de toekomst een bijdrage kunnen leveren aan de testbank.

Uitgevoerd in 2022

- Nieuwe omgeving TeamCity ingericht voor iMOD5.
- Bestaande regionale test(deel)modellen omgezet van iMODFLOW naar MODFLOW 6 en toegevoegd aan TeamCity.
- MODFLOW6 –MetaSWAP case toegevoegd aan TeamCity (Brabant Delta)
 - Uitbreiding van MetaSWAP case met variant inclusief WOFOST
 - iMOD Coupler tests met MODFLOW6–MetaSWAP case
 - iMOD Preprocessing tests met MODFLOW6–MetaSWAP case
- Serie iMOD Batch functies voor pre- en postprocessing toegevoegd, bijvoorbeeld IDFTIMESERIES.
- Testmodellen uit iMOD Tutorials (zie manual) toegevoegd als case aan TeamCity waaronder de iMOD-WQ varianten MT3D en SEAWAT
- In het kader van TKI ontwikkelingen een testcase voor de nieuwe MetaSWAP module toegevoegd: Regelbare Drainage en Infiltratie.

2 iMOD modelcodes en bijbehorende testomgeving

2.1 Overzicht en toelichting

Onderstaande tabel beschrijft voor elke modelcode aan de hand van een aantal karakteristieken hoe het testproces is ingericht. Er is onderscheid tussen MODFLOW en MetaSWAP gerelateerde rekencodes en software voor preprocessing en viewing. Alleen beheer en onderhoud van iMOD-GUI, iMODFLOW en MetaSWAP vallen onder het NHI project “Beheer en Onderhoud modelcodes”.

	Modelcodes: MODFLOW6	iMODFLOW/ iMODFLOWMetaSWAP	MetaSWAP	iMOD Coupler MF6-MS	iMOD-WQ	iMOD-GUI	Preprocessing: iMOD-batch	iMOD-Python
Aanspreekpunt	USGS	Peter V.	Paul vW., Joachim H.	Martijn R.	Gijs J.	Peter V.	Peter V.	Huite B.
Testomgeving	Github	Python + DOS	TeamCity (TC) + DOS	TeamCity	Python	Team van tester	handmatig	Gitlab
Testfrequentie	Continu	Release + tussenversies	Release + tussenversies + continu (TC)	continu	Release	Release	Release	Continu
Test cases	<i>Functietesten</i>	<i>Functietesten</i> <i>Deelmodellen:</i> Moria / Ibrahim Azure / Mipwa / LHM MF	<i>Functietesten</i> <i>Model:</i> Doorgaans LHM LHM Light	<i>Functietesten:</i> o.a. Stripmodel, inactieve cellen, berekening	<i>Functietesten:</i> Henry Case / Wells <i>Modellen:</i> DOW Terneuzen, A'foort, Europort	Tutorials	Naar TeamCity	<i>Functietesten</i> <i>Deelmodellen:</i> Mipwa Tutorials
Rapportage	Automatisch Testrapport	Tabel / PNG	Testrapporten veranderingsvo orstellen LHM rapportage	Automatisch Testrapport	Tabel / PNG	-	-	Automatisch Testrapport

Figuur 2.1: overzicht van Modelcodes en bijbehorende testprocessen

Bij deze tabel zijn enkele opmerkingen te maken:

MODFLOW6

Deze software test Deltares niet. Zie toelichting in paragraaf 2.2.

iMODFLOW / iMODFLOWMetaSWAP

Deze ontwikkel- en testomgeving wordt geleidelijk uitgefaseerd. iMODFLOW2005 wordt niet meer ondersteund en het advies is om over te stappen naar MODFLOW6. Deltares richt alle nieuwe ontwikkelingen op MODFLOW6.

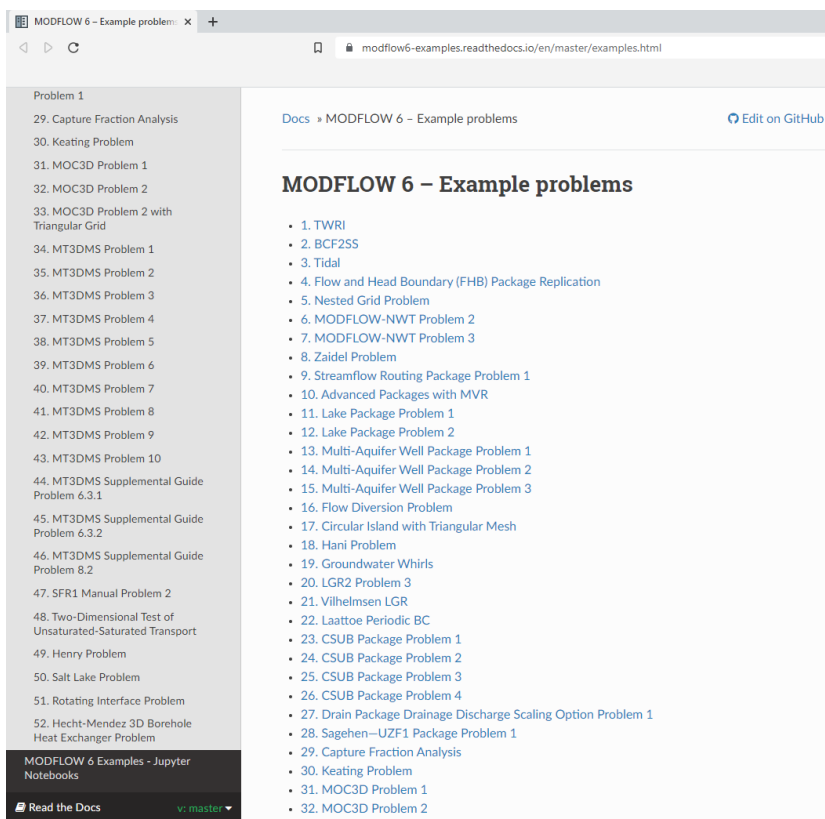
- **MetaSWAP:** Ontwikkeling van MetaSWAP is/wordt gerealiseerd door WENR/VanWater. In samenwerking tussen VanWater en Deltares is een proces ingezet waarbij het beheer en onderhoud verder wordt gestructureerd bij TeamCity als testomgeving gebruikt wordt.
- **iMOD Coupler:** Dit is de jongste modelcode in de iMOD familie. Ontwikkeling en testen vindt vanaf de eerste versie volledig plaats op de TeamCity omgeving.
- **iMOD WQ:** de methode van ontwikkelen en testen heeft veel gelijkenis met iMODFLOW. De hoeveelheid bugfixes en feature toevoegingen/wijzigingen is beperkt.
- **iMOD5 GUI:** er bestaan uitgebreide tools om ook grafische userinterfaces te testen. Het vergt te veel investering om op dit moment iMOD5 geschikt te maken voor dergelijke automatische testsoftware. De GUI wordt daardoor doorgaans getest door een testteam van studenten en junior collega's.
- **iMOD5 Batch:** het is een nieuwe stap om zoveel mogelijk batch functies aan TeamCity toe te voegen. Tot nu toe werden deze functies meegetest bij het testen van de iMOD GUI. Belangrijkste functie is om de preprocessing te verzorgen voor MODFLOW6-MetaSWAP modellen. Die testbank gebruikt het platform TeamCity.
- **iMOD-python:** deze tool wordt op Github ontwikkeld en dit platform ondersteunt ook het testproces.

2.2 Wat wordt niet getest?

Deltares test wel iMODFLOW maar MODFLOW6 staat niet in het overzicht. Daar is een reden voor: iMODFLOW was een Deltares kloon van MODFLOW2005 waarin aan de code zelf veel functies (waaronder volledige preprocessing en modules) waren toegevoegd. Die aanpassingen moest worden getest in de testbank.

Bij de komst van MODFLOW6 heeft Deltares besloten om geen eigen versie van MODFLOW meer te maken maar de aanpassingen direct in de kern van de rekencode van de USGS te ontwikkelen, in samenwerking met de USGS. Daarom wordt vooraan alleen de originele USGS versie van MODFLOW6 gebruikt. De kwaliteitsborging is onder verantwoording van de USGS.

De USGS heeft een MODFLOW6 testomgeving (<https://github.com/MODFLOW-USGS/modflow6>) die toegankelijk is voor iedereen. De testfrequentie is continu. Het aantal tests is op dit moment groter dan 500 en deze zijn volledig beschreven op deze website: <https://modflow6-examples.readthedocs.io/en/master/examples.html>. Testrapporten en logfiles zijn online beschikbaar.



Figuur 2.2: Website met alle beschikbare testvoorbeelden voor MODFLOW6

3 Nieuwe testomgeving: TeamCity

Belangrijkste stap bij de uitbreiding van de testbanken is de inrichting en het gebruik van de nieuwe TeamCity testomgeving. Dit hoofdstuk geeft een korte toelichting op de werking ervan.

3.1 Onderdelen testproces

TeamCity is een serveromgeving waarop zogenaamde “continuous integration tests” worden uitgevoerd. Het programma is ontwikkeld door JetBrains en meer informatie is te vinden via <https://www.jetbrains.com/teamcity/>

TeamCity werkt als volgt. Zodra een ontwikkelaar een update gedaan heeft van een rekencode op version control (SVN) start TeamCity automatisch een serie processen. Voor iMOD 5 zijn dat de volgende stappen (zie figuur Figuur 3.1).

Fase 1: Build

TeamCity haalt alle iMOD5 code binnen vanuit SVN. Hiervoor kiest TeamCity een vrij rekenknoop (agent) op het systeem. Het systeem compileert de Fortran code en levert een Executable op. Mocht er een fout ontstaan bij het compileren, dan krijgt het ontwikkelteam daar een melding van.

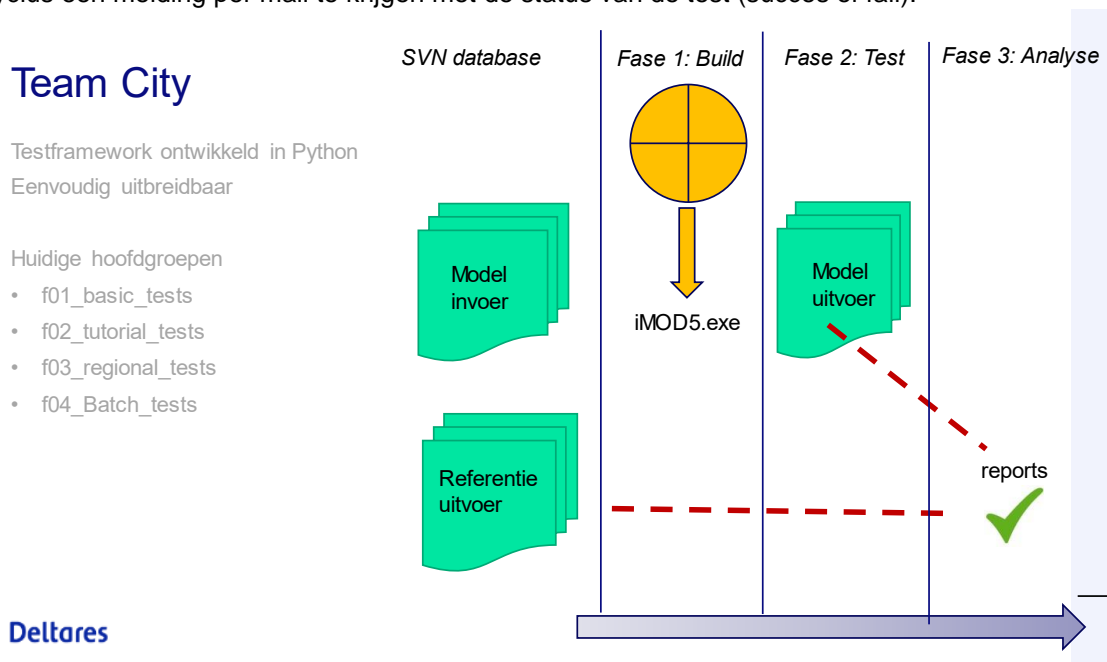
Fase 2: Test

TeamCity start het proces waarbij alle beschikbare tests worden gedraaid. In het geval van MODFLOW en MetaSWAP worden testmodellen gedraaid. Voor het testen van iMOD Batch functies zijn er specifieke functietesten. Alle testen leveren uitvoer in de vorm van IDF's en IPF's of tekstbestanden.

Fase 3: Analyse

In de laatste fase worden de uitkomsten vergeleken met een standaardset die vooraf is vastgesteld. De vergelijking kan bestaan uit verschilgrids van twee IDF's of een vergelijk op inhoud van tekstbestanden. Die vergelijking kan grote, kleine of geen verschillen opleveren. Het is aan de ontwikkelaar om te bepalen hoe strenge de analyse moet zijn, welke verschillen acceptabel zijn en welke moeten leiden tot een melding van het falen van een test.

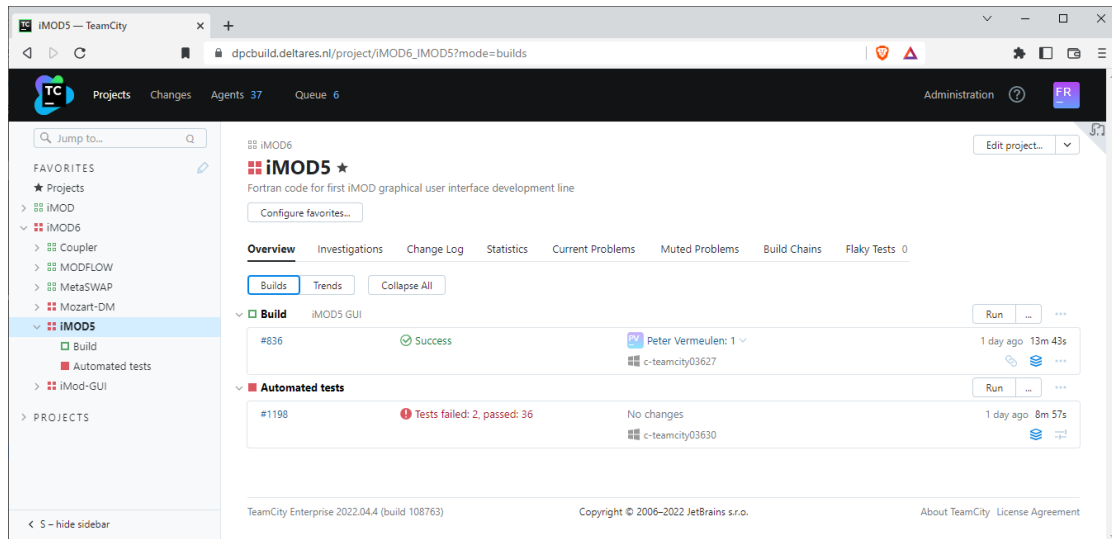
Het ontwikkelteam kan de processtappen eventueel live volgen. Er is de optie om na elke testcyclus een melding per mail te krijgen met de status van de test (succes of fail).



Figuur 3.1: Schematisch testproces op TeamCity

Op dit moment is alleen een update van iMOD5 een trigger om het testproces te starten. Het is ook mogelijk om een start te doen bij een nieuwe versie van de iMOD-Coupler of

MetaSWAP. Op dit moment gebruikt TeamCity voor Coupler en MetaSWAP de laatst gereleasde versie



Figuur 3.2: TeamCity Dashboard voor testproces van iMOD5 met fase 1 (Build) en fase 2 en 3 (Test/Analyse)

Er zijn veel voordelen van het gebruik van TeamCity. Belangrijk is dat de tests automatisch worden uitgevoerd na elke update. Dat versnelt het integratie proces richting een nieuwe release en maakt het overzichtelijk om dat het stap voor stap gaat. De rapportage is uitgebreid en het hele team is op de hoogte van de performance van de testen.

Niet alleen iMOD wordt op TeamCity getest. Ook ander Deltares producten draaien op deze omgeving zoals D-HYDRO, DGEO, DGFLOW, WANDA.

3.2 Opbouw en analysemogelijkheden

De TeamCity testbank bestaat uit 3 onderdelen:

- een set scripts waarin de tests worden beschreven. De aansturing is volledig in Python geschreven en maakt gebruik van de module Pytest.
- Een dataset met voor elke testcase de benodigde invoerdata. Voorbeelden zijn een PRJ file met alle modelinvoer parameters, een INI file met alle run opties en alle invoerbestanden zelf.
- Een dataset met voor elke testcase de benodigde referentie (uitvoer) bestanden. Deze benchmark wordt vergeleken met de testuitvoer en bepaalt of de tests al of niet falen

Het gebruik van Python module Pytest geeft de gebruiker volledige vrijheid om het testen en analyseren in te richten. Omdat de tests op elkaar lijken, zijn er enkele standaard Python functies gedefinieerd wat het eenvoudig maakt om nieuwe tests toe te voegen.

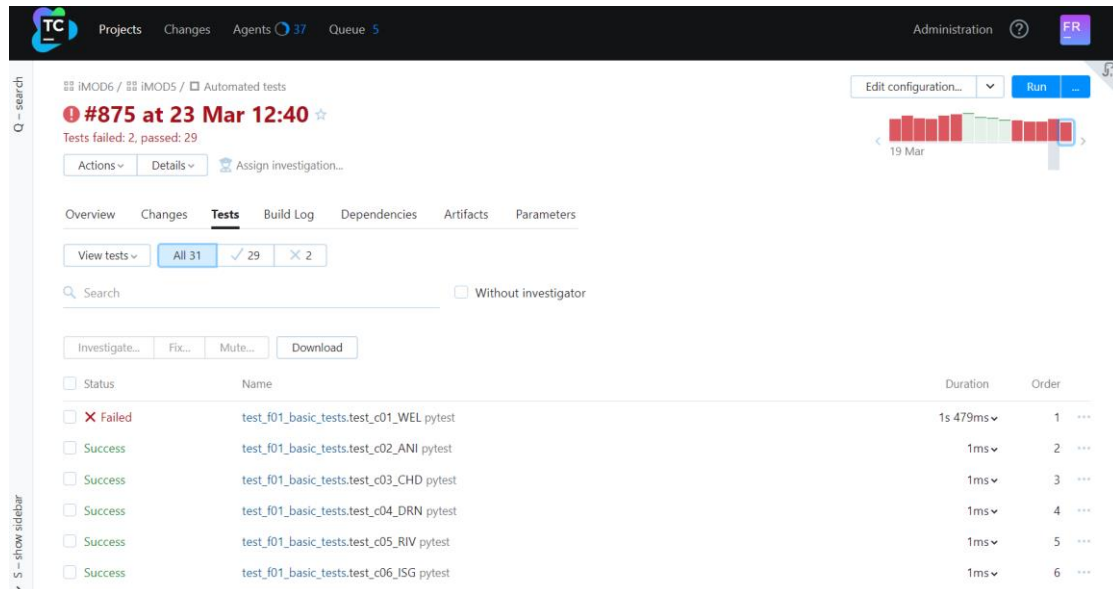
3.2.1 Overzicht testgroepen

De nieuwe testset voor de MODFLOW6 preprocessing met iMOD5 is momenteel onderverdeeld in 5 groepen.

- **Basic Tests**
Daarin is voor elk MODFLOW package een of meer testen opgenomen. Denk aan onttrekkingen (WEL), waterlopen (RIV en DRN) en grondwateraanvulling (RCH).

- Tutorial tests

Testmodellen uit iMOD Tutorials (zie manual) toegevoegd als case aan TeamCity waaronder de iMOD-WQ varianten MT3D en SEAWAT.



Figuur 3.3: Team City dashboard met overzicht succesvolle en falende tests

- Regionale tests

Deelgebieden uit de regionale modellen zoals Ibrahim, Moria en Mipwa. Een deelgebied van Brabantse Delta is in het kader van een project ook toegevoegd.

- iMOD Batch functietests

Het betreft hier tests voor de meest gebruikte iMOD Batch functies zoals IDFGXG, IDFMATH, IDFCALC, IPFSAMPLE, IDFTIMESERIES en MF6TOIDF.

- Test voor iMOD Coupler.

Hier testen we de combinatie van MetaSWAP en MODFLOW6.

3.2.2 Test analyse overzichten

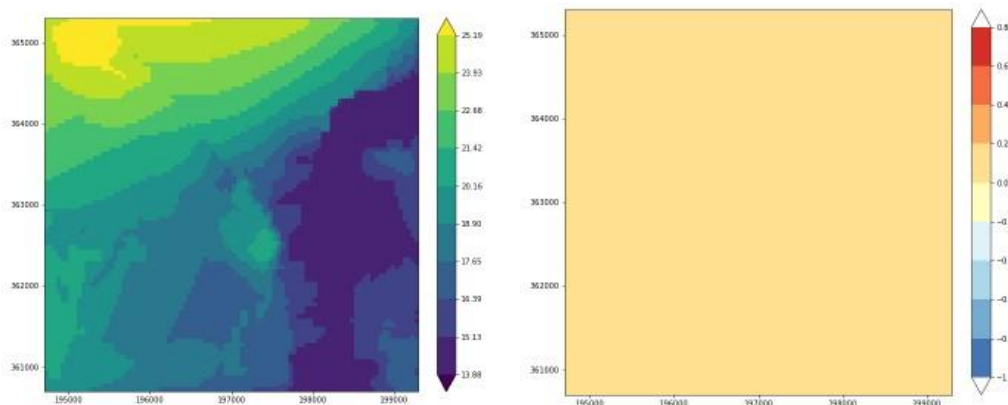
De uitkomsten van elke testcyclus op TeamCity kan online bekeken worden. Dit is echter een analyse op abstract niveau; welke testen hebben gefaald en zo ja op welke punten. Vanuit het oogpunt van een hydroloog is juist nuttig om op een dieper niveau te kijken. Bijvoorbeeld om te zien op welke plekken of momenten in het model de afwijkingen t.o.v. de benchmark zijn opgetreden. Die post-processing op de testsuitkomsten kan door de ontwikkelaar volledig zelf worden ingericht.

Op dit moment wordt geanalyseerd op 2 punten:

- Zijn de IDF uitvoerbestanden gelijk, denk bijvoorbeeld aan de berekende HEAD. Daarvoor wordt een PNG gemaakt van zowel de modeluitkomst als het verschil met de benchmark (zie Figuur 3.4).
- Is de inhoud van tekstbestanden gelijk? Dat betreft bestanden zoals *.IPF (en associated txt files) maar ook *.CSV.

De samenvatting van beide analyses is terug te vinden in een LOG file zodat snel gezien kan worden in welke bestand de afwijking zit.

_RESULTS_GWF_1_MODELOUTPUT_HEAD_HEAD_HEAD_STEADY-STATE_L1_case.PNG



Figuur 3.4: Voorbeeld Team City testuitvoer – ruimtelijke uitvoer: IDF uitvoer van Heads (links) vergeleken met de referentie (rechts verschilkaart)

Het testcriterium is op dit moment erg streng: zodra er ook maar een minimale afwijking wordt geconstateerd, faalt de test.

3.3 Eerste ervaringen in de praktijk

Wat gebeurt er nu als er een test faalt?

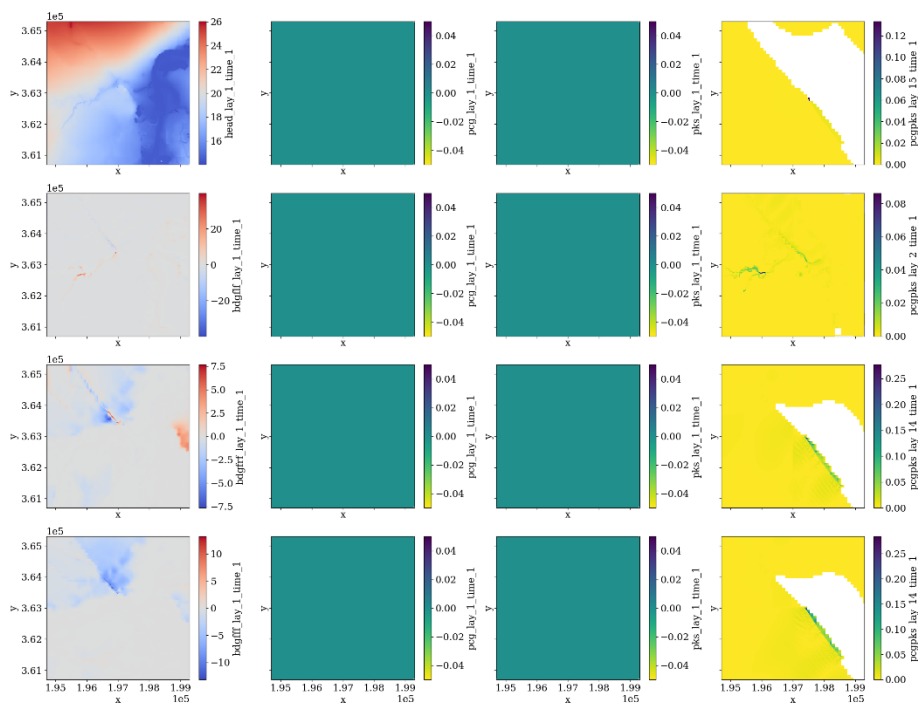
Na afloop van een testcyclus geeft TeamCity een duidelijke melding welke individuele tests zijn gefaald. Het is nu zaak om de reden te achterhalen. De volgende stap is dan ook dat de ontwikkelaar betreffende tests opnieuw draait om de fout te kunnen reproduceren. Dat gebeurt binnen een kopie van de testomgeving op de eigen machine. Op die manier is het mogelijk om nog meer analyses uit te voeren waaronder het uitlezen van de MODFLOW Log file. Als de oorzaak van de afwijking is gevonden, wordt er een beslissing genomen:

- De afwijking had niet mogen optreden. De software moet worden op dit punt worden aangepast net zo lang tot dat de test niet meer faalt.
- De afwijking is verklaarbaar/gewenst door aanpassingen in de software. De benchmark van de test wordt aangepast aan de nieuwe situatie. De test faalt niet meer.

Benchmark

Het voordeel van deze aanpak is dat bij elke *commit* van een nieuwe versie van de code goed naar de uitkomsten wordt gekeken. Het nadeel van aanpak, naast dat het veel tijd kost, is dat er een kleine kans bestaat dat een fout over het hoofd wordt gezien en wordt geaccepteerd. Deze komt dan ongezien in de benchmark terecht.

Om dit te ondervangen overwegen we om in de aanloop naar een release de software te vergelijken met een benchmark van de vorige gereleasde versie. Het verschil dat dan zichtbaar wordt, is een combinatie van alle effecten van wijziging in de code. Op die manier is de huidige testbank voor iMODFLOW ook ingericht.



Figuur 3.5: Voorbeeld uitvoer testbank iMODFLOW. Ruimtelijke verschil berekend m.b.v. laatste gereleasde versie en huidige versie.

Type fouten onderscheiden

In de praktijk blijkt het ingewikkeld te zijn om snel te zien welk product de oorzaak is van een test failure. Dat komt door de huidige opbouw van de tests. Die analyseren alleen de verschillen in de einduitkomsten (bijv. IDF met Heads) en niet de tussenproducten. Maar de fout kan ook in (tussenproducten van) andere componenten zitten:

- is het de preprocessing met iMOD5 die de MODFLOW invoer maakt (INP bestanden)
- is het de berekening door MetaSWAP
- is het de uitwisseling van data via de iMOD-Coupler
- is het een wijziging in MODFLOW6

Om dit te verbeteren wordt aanbevolen dit op te pakken in het kader van BenO 2023. Dat kan door de testen te verfijnen en in de analyse ook tussenproducten te gaan testen of door de afzonderlijke producten apart te testen.

4 Toekomst iMOD testbanken

4.1 Betrokkenheid derden

De kern van de ontwikkeling van de iMOD gerelateerde tools ligt bij Deltares. In het kader van programma's als NHI en TKI is er samenwerking met andere partijen bij de ontwikkeling van nieuwe onderdelen. Het Deltares ontwikkelteam voert de daadwerkelijke implementatie van de aanpassingen door in de modelcode.

Deltares heeft een voorstel gedaan op welke wijze marktpartijen en andere partijen betrokken zouden kunnen zijn bij beheer en onderhoud van de testbanken. Partijen kunnen een aantal testmodellen samenstellen. Bij de testmodellen kunnen testcriteria worden opgesteld, uitgewerkt in een serie Python scripts. De modellen en tests moeten ontwikkeld zijn voor MODFLOW6 en beschikbaar zijn in iMOD formaat en aangestuurd via een PRJ file. Creativiteit is ook nuttig bij de doorontwikkeling van de testbanken. Tot slot kunnen partijen mogelijk een rol spelen bij de evaluatie van de testrapporten.

4.2 Beheer & Onderhoud: conclusie en aanbeveling

Het testen van iMOD software is altijd onderdeel geweest van het ontwikkelproces. Met de overgang naar continue integratietesten op TeamCity is een grote professionaliseringsstap gezet. TeamCity is vanaf nu toegevoegd als onderdeel aan Beheer en Onderhoud van de software.

Deze stap is het begin. Voor de toekomst adviseren we het volgende:

- Nieuw ontwikkelde modules worden toegevoegd aan de testbank. Een recent voorbeeld hiervan is de ontwikkeling van de module Regelbare Drainage en Infiltratie in MetaSWAP (release iMOD5.4). Binnenkort verwachten we de module bodemdaling toe te voegen.
- Bugs in iMOD worden opgelost en in de volgende iMOD release uitgeleverd. Als het toegevoegde waarde heeft kan voor een bug een test worden toegevoegd aan de testbank.
- De modelcode voor MetaSWAP wordt buiten TeamCity ontwikkeld en getest. Onlangs is MetaSWAP wel toegevoegd aan TeamCity en wordt daar gecompileerd. Er draait een kleine set testen voor MetaSWAP. Het advies is om deze testen uit te breiden in 2023.
- Op dit moment wordt de combinatie *iMOD5 preprocessing – MetaSWAP – iMOD Coupler op TeamCity* getest. De testset bestaat uit 2 tests op basis van deelmodellen uit het model voor waterschap Brabantse Delta. Voor een meer complete testset wordt uitbreiding aanbevolen, dat kan worden opgepakt in het kader van Beheer en Onderhoud 2023.