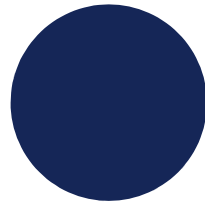


# *Einführung in die Modellierung*

Till Francke und Maik Heistermann  
*Universität Potsdam*



Seminar *Einführung in die Modellierung*  
im Modul *Versuchsplanung und Geoökologische Modellierung*

# ***Einführung in die Modellierung***

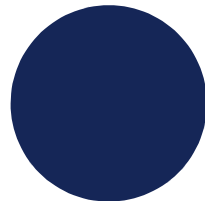
**In diesem Semester**

R als Werkzeug in der Modellierung

Ökologische Modelle

**Hydrologische Modelle**

(Ökohydrologische Modelle)



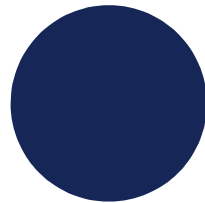
# ***Einführung in die Modellierung***

**Heute**

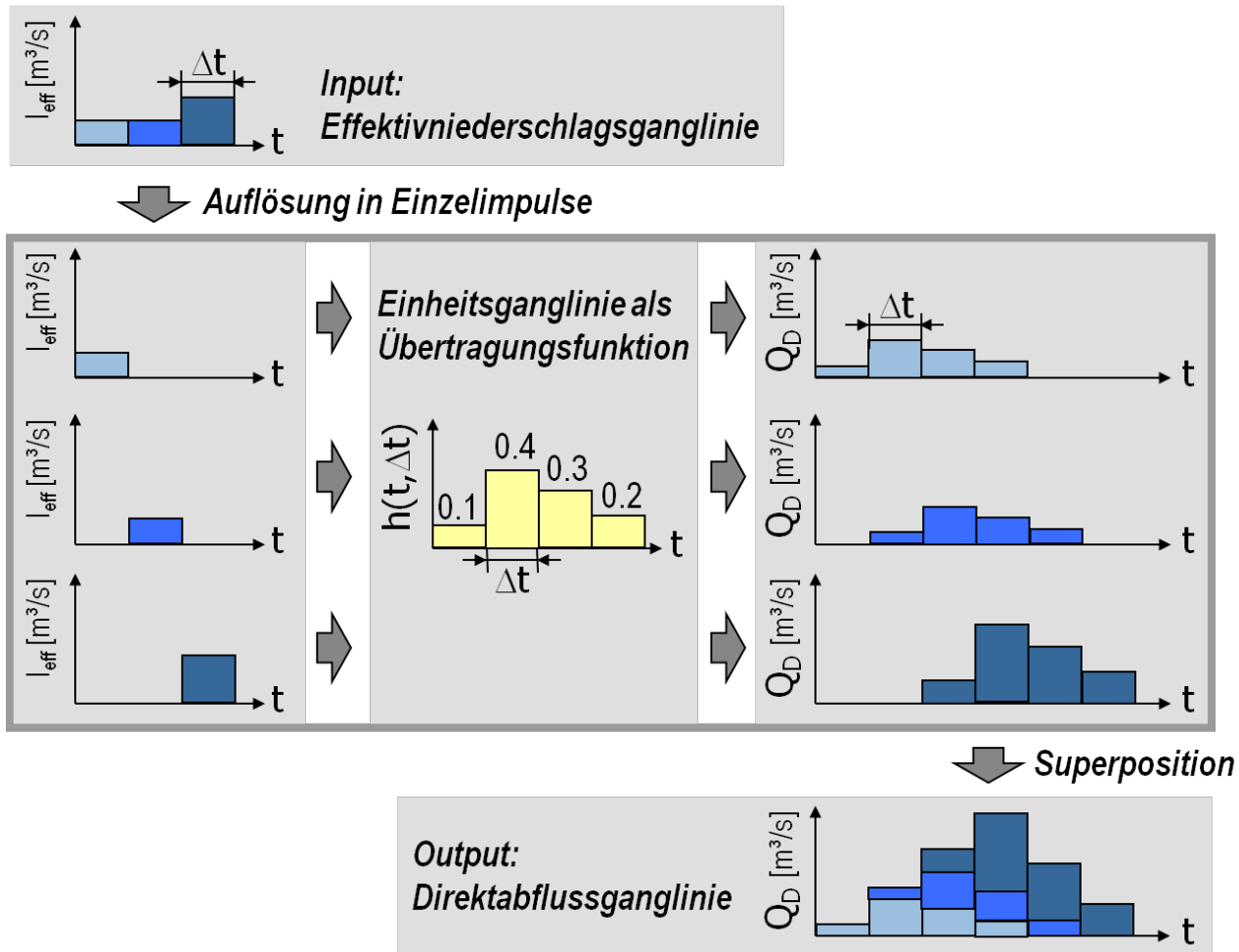
Die Einheitsganglinie

Das *abc* der Wasserhaushaltsmodellierung

Routinekämpfe



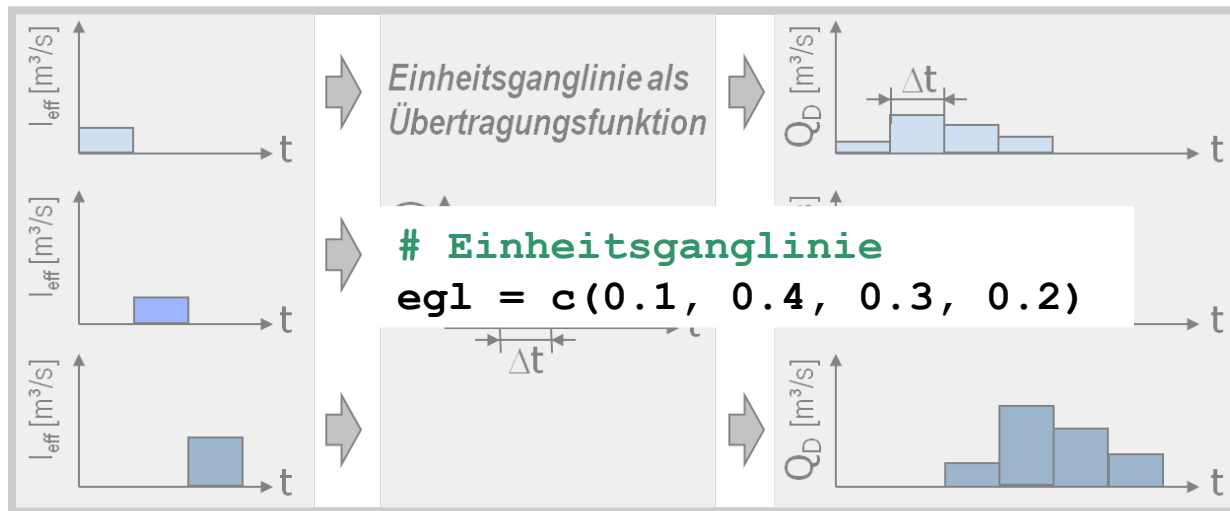
## 2. Semester... Hydro... Einheitsganglinie ohne Ende...



## Programmatische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)
```

↓ Auflösung in Einzelimpulse



↓ Superposition

```
# Abflussganglinie (in m3/s) - „Container“
out = rep(0., length(precip) + length(egl) - 1)
```

## Programmatische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)

# Einheitsganglinie
egl = c(0.1, 0.4, 0.3, 0.2)

# Abflussganglinie (in m3/s) - Container
out = rep(0., length(precip) + length(egl) - 1)

# Fläche des Einzugsgebiets (km2)
area = 10
```

## Programmatistische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)

# Einheitsganglinie
egl = c(0.1, 0.4, 0.3, 0.2)

# Abflussganglinie (in m3/s) - Container
out = rep(0., length(precip) + length(egl) - 1)

# Flaeche des Einzugsgebiets (km2)
area = 10

# Schleife über alle Niederschlagsimpulse
for (i in 1:length(precip)) {
```

```
}
```

## Programmatistische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)

# Einheitsganglinie
egl = c(0.1, 0.4, 0.3, 0.2)

# Abflussganglinie (in m3/s) - Container
out = rep(0., length(precip) + length(egl) - 1)

# Flaeche des Einzugsgebiets (km2)
area = 10

# Schleife über alle Niederschlagsimpulse
for (i in 1:length(precip)) {
  # Abflussantwort auf Impuls i
  out_i = precip[i] * egl * area / 3.6
  # Zeitindices, auf welche sich Impuls i verteilt
  ix = i:(i+length(egl)-1)
  # Addiere auf Abflussganglinie
  out[ix] = out[ix] + out_i
}
```



## Programmatische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)

# Einheitsganglinie
egl = c(0.1, 0.4, 0.3, 0.2)

# Abflussganglinie (in m3/s)
out = rep(0., length(precip) + length(egl) - 1)

# Fläche des Einzugsgebiets (km2)
area = 10

# Schleife über alle Niederschlagsimpulse
for (i in 1:length(precip)) {
  ...
}

# Resultierende Abflussganglinie (in m3/s)
> print(out)
[1] 0.2 1.8 6.6 11.4 8.0 4.0
```

## Programmatische Formulierung des EGL-Verfahrens

```
# Effektivniederschlagsganglinie (in mm/h)
precip = c(1., 5., 10.)

# Einheitsganglinie
egl = c(0.1, 0.4, 0.3, 0.2)

# Abflussganglinie (in m3/s)
out = rep(0., length(precip) + length(egl) - 1)

# Fläche des Einzugsgebiets (km2)
area = 10

# Schleife über alle Niederschlagsimpulse
for (i in 1:length(precip)) {
  ...
}
```



EGL als Funktion schreiben (einheitsganglinie.R)

Annahme eines konstanten Basisabflusses einführen (einheitsganglinie.R)

Zeitreihen als Diagramm visualisieren (einheitsganglinie.R)

## Einige Schwächen des Einheitsganglinienverfahrens

- Quantifizierung der Abflussbildung (Abflussbeiwert)?
- Dynamik des Basisabflusses wird ignoriert
- Keine Massenerhaltung im Sinne einer Gebietswasserbilanz
- Black-Box-Transformation von Niederschlag in Abfluss

## Einige Schwächen des Einheitsganglinienverfahrens

- Quantifizierung der Abflussbildung (Abflussbeiwert)?
- Dynamik des Basisabflusses wird ignoriert
- Keine Massenerhaltung im Sinne einer Gebietswasserbilanz
- Black-Box-Transformation von Niederschlag in Abfluss

## Alternative: Abfluss als Komponente des Wasserhaushalts

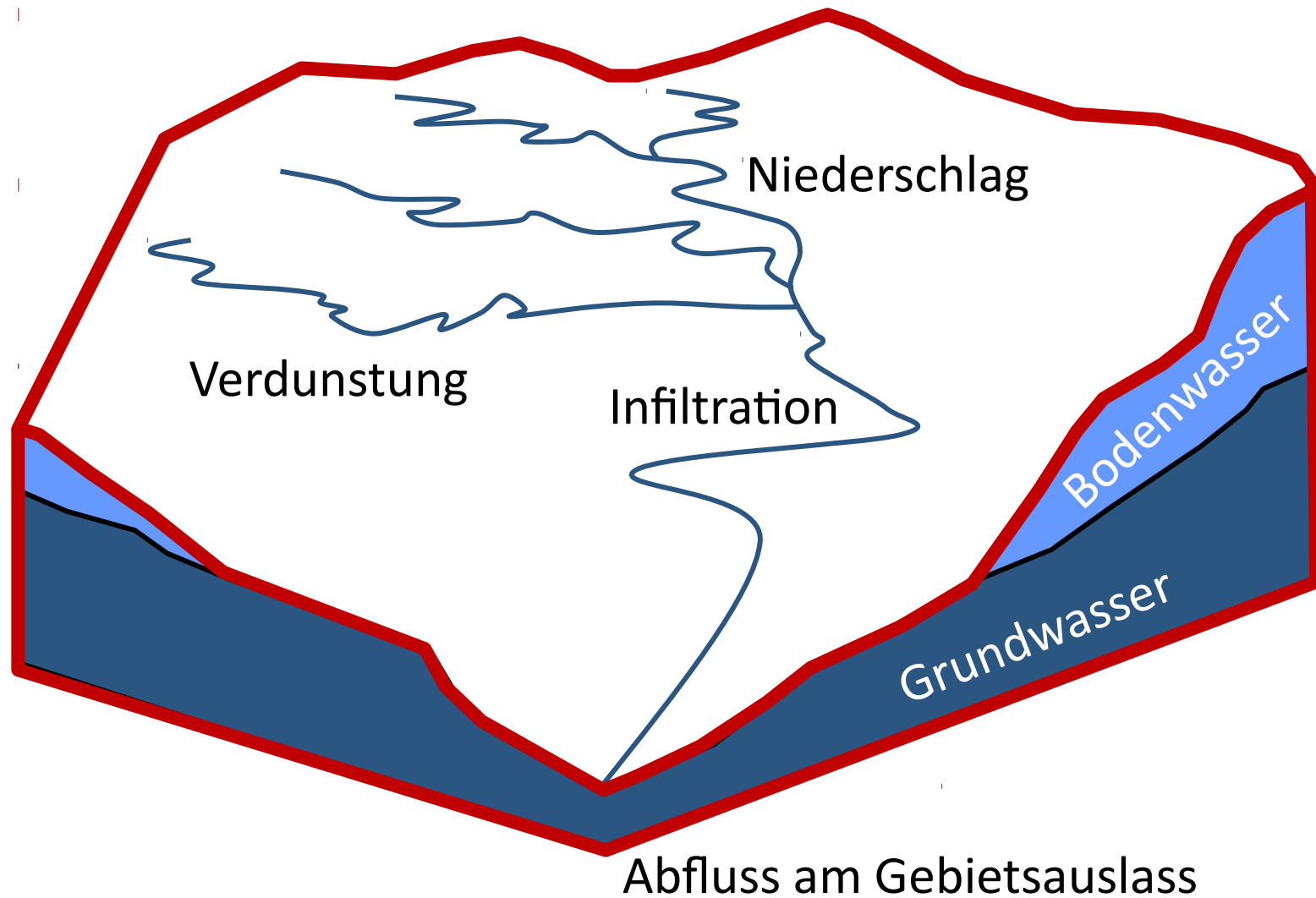
$$\frac{\Delta S}{\Delta t} = P - ET - Q$$

- ⇒ Massenerhaltung als Grundprinzip
- ⇒ Einfluss der Gebietsfeuchte auf Abflussbildung
- ⇒ Abbildung physikalischer Prozesse (z.B. Verdunstung, Schnee)

# Wasserhaushaltsmodelle



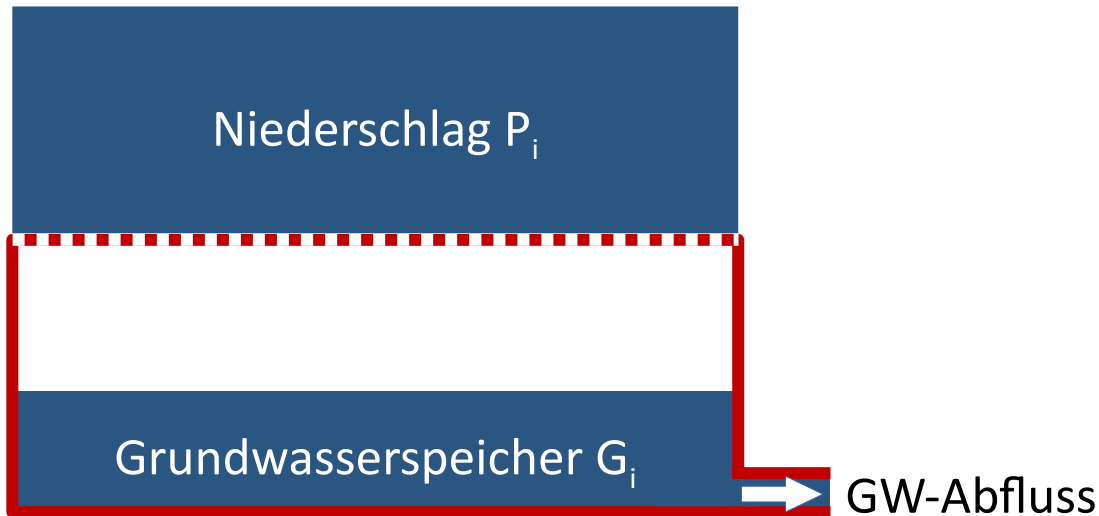
# Wasserhaushaltsmodelle



## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell

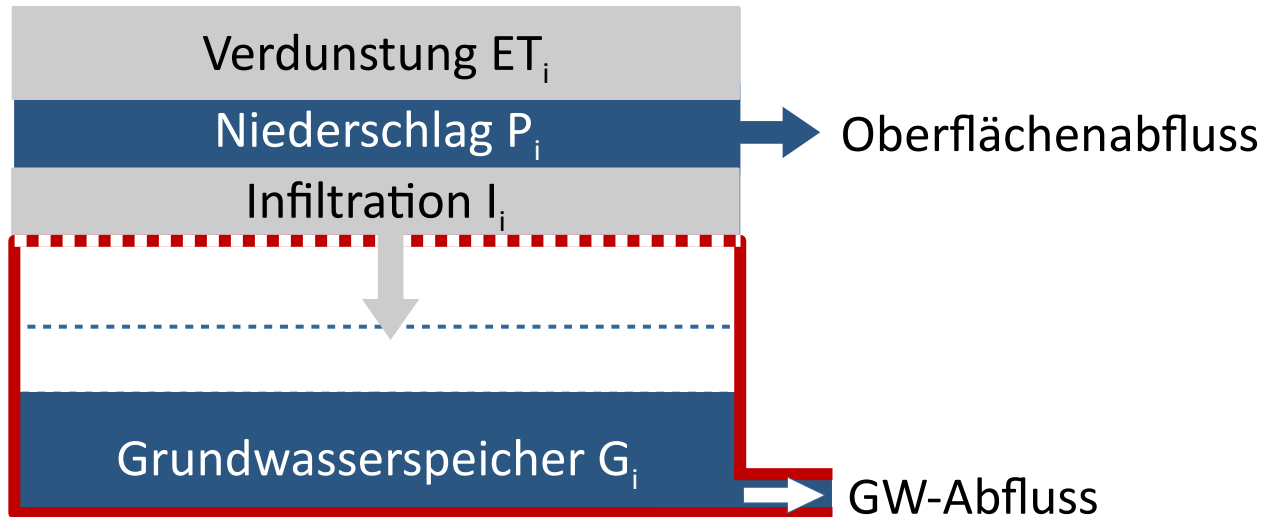


## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell

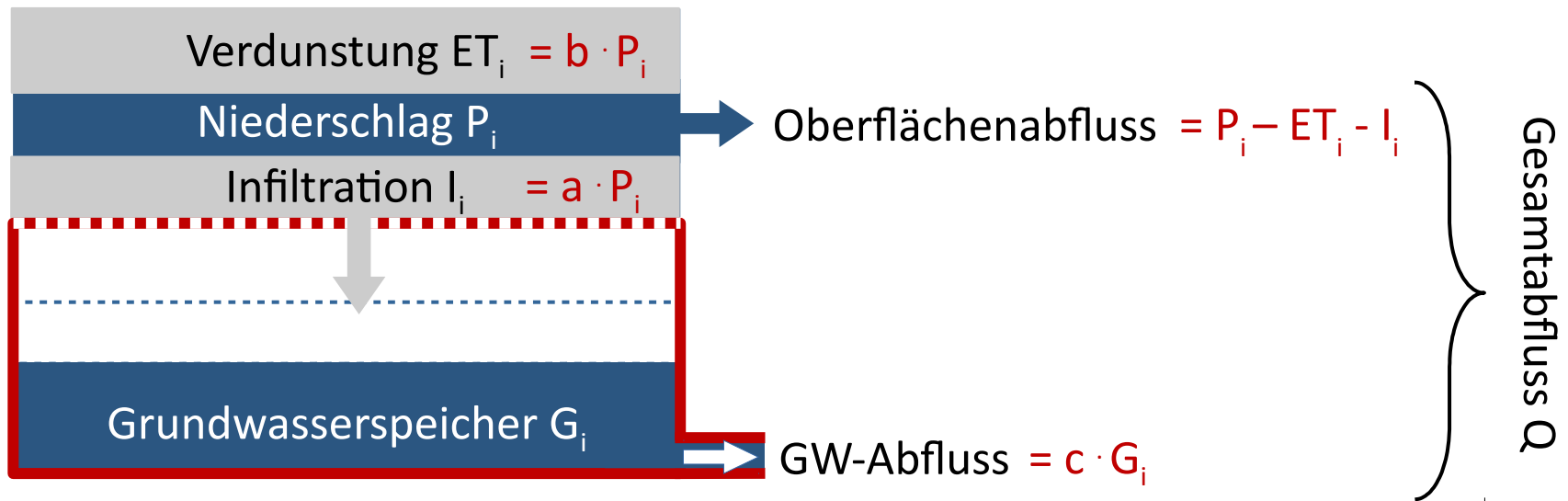




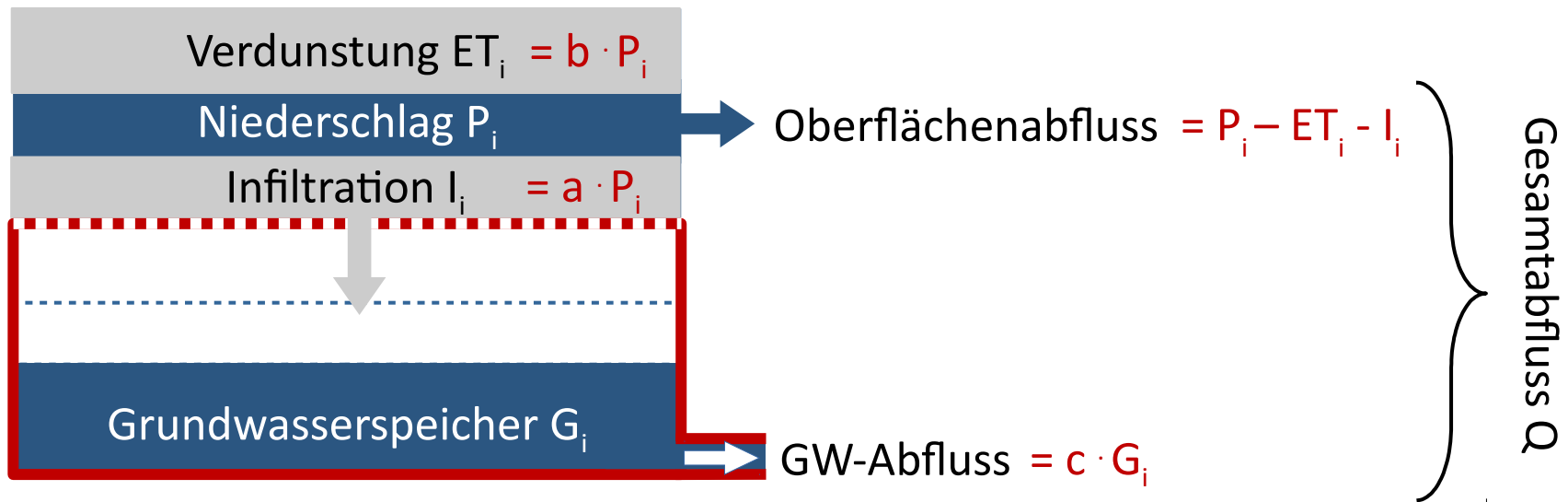
## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell

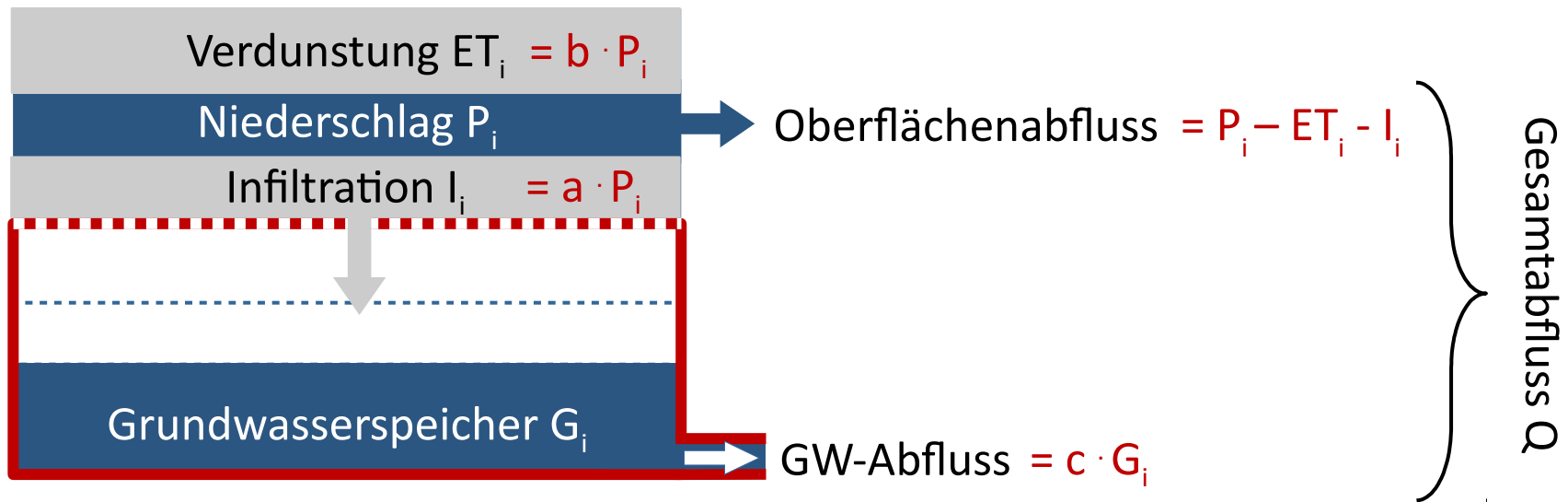


## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



Wie lauten die Einheiten für diese Größen?

## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



### Wie lauten die Einheiten für diese Größen?

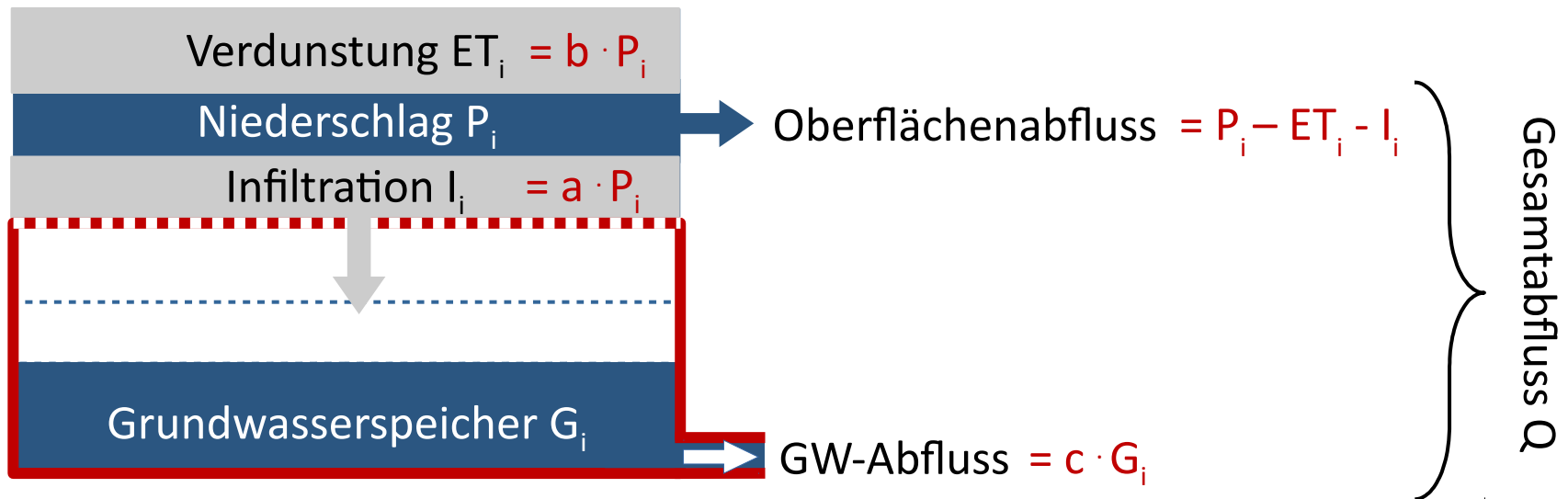
[L/T] oder [V/T] für  $Q$ ,  $ET$ ,  $P$ ,  $I$

[L] oder [V] für  $G$

[1/T] für  $c$

[-] für  $a$ ,  $b$

## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell

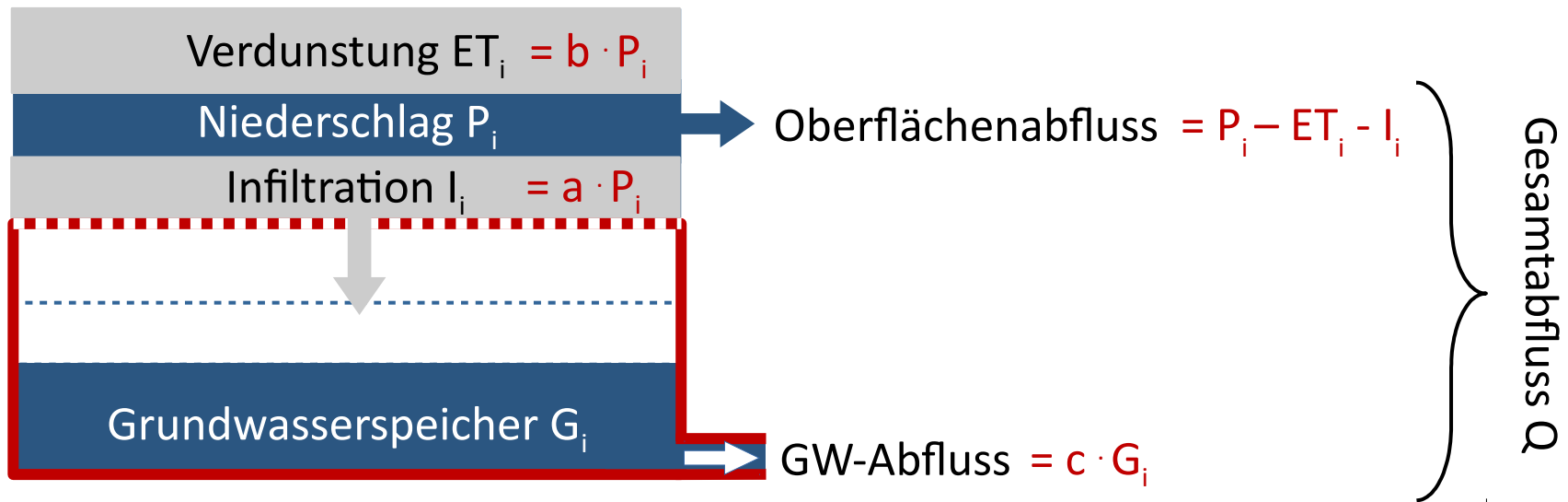


**Gleichung für Gesamtabfluss  $Q_i$  am Gebietsauslass**



**Neuberechnung des Grundwasserspeichers  $G_i$  im Zeitschritt  $i$**

## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



**Gleichung für Gesamtabfluss  $Q_i$  am Gebietsauslass**

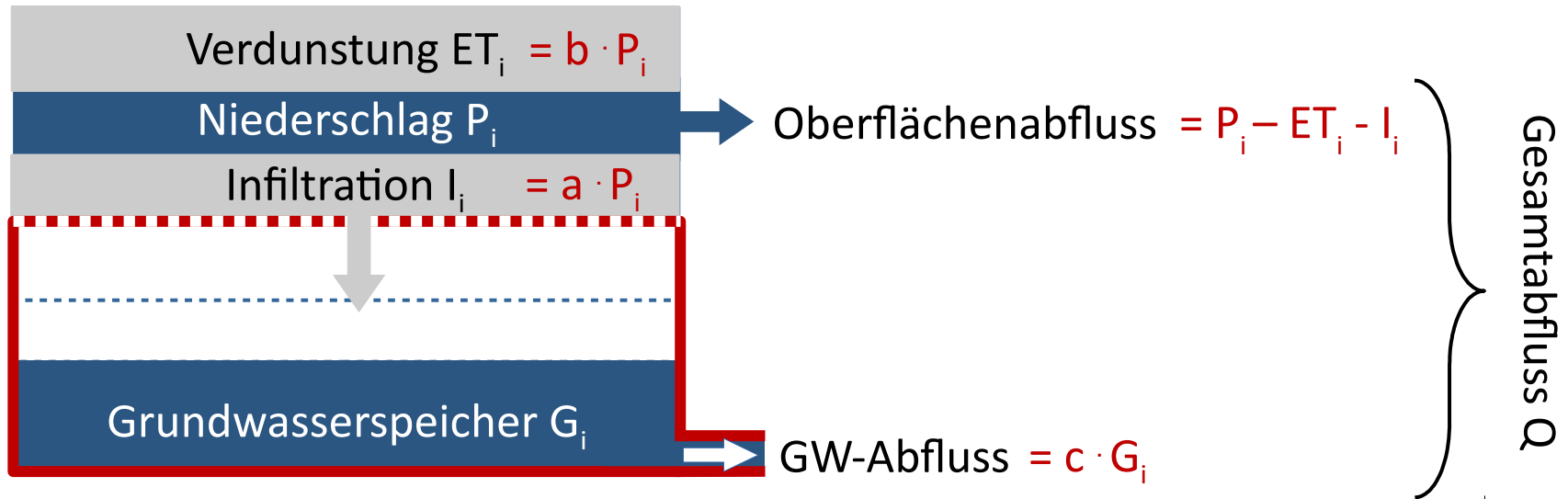
$$Q_i = AO_i + AG_i = (1-a-b)P_i + cG_{i-1}$$



**Neuberechnung des Grundwasserspeichers  $G_i$  im Zeitschritt  $i$**

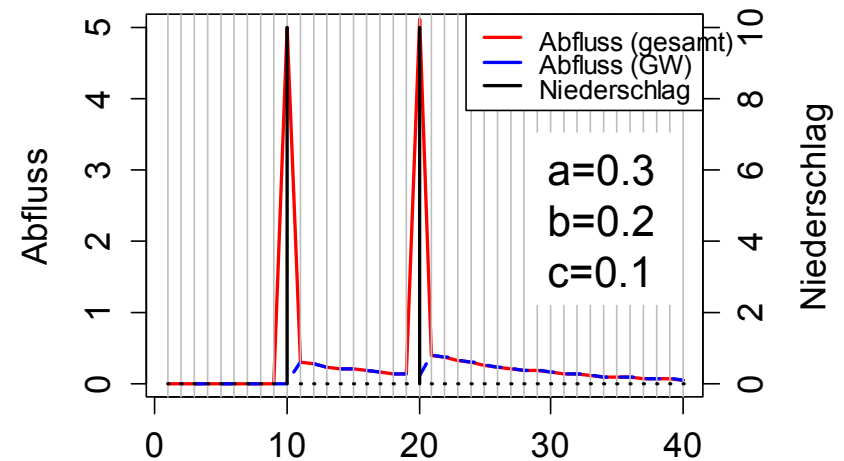
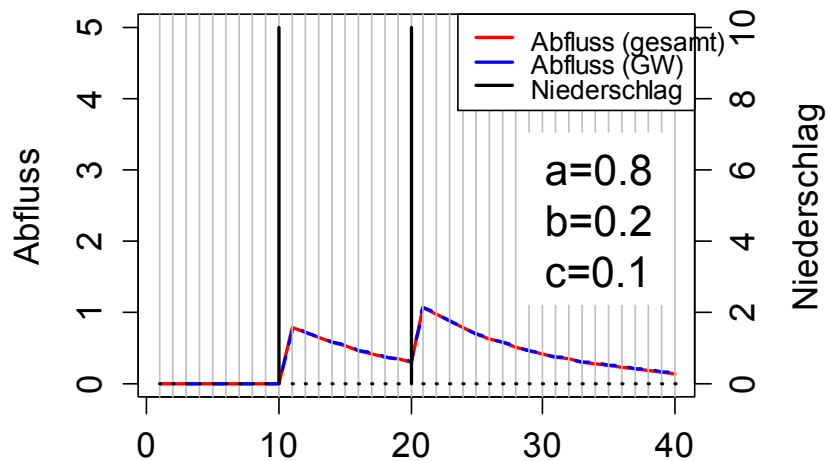
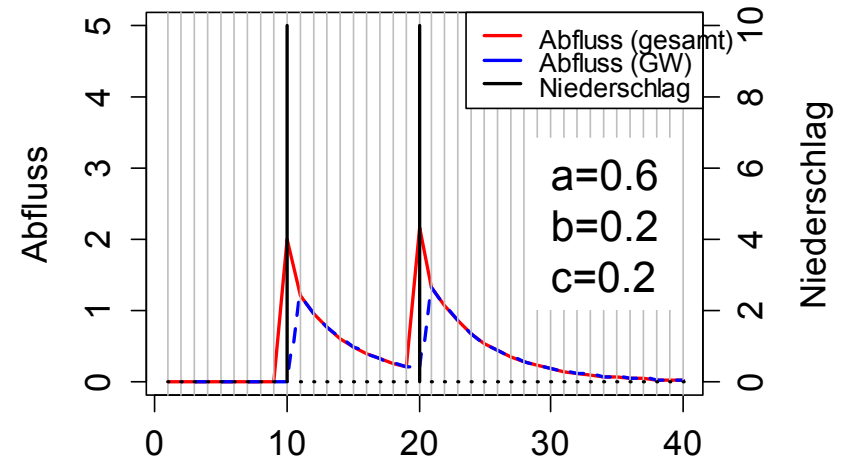
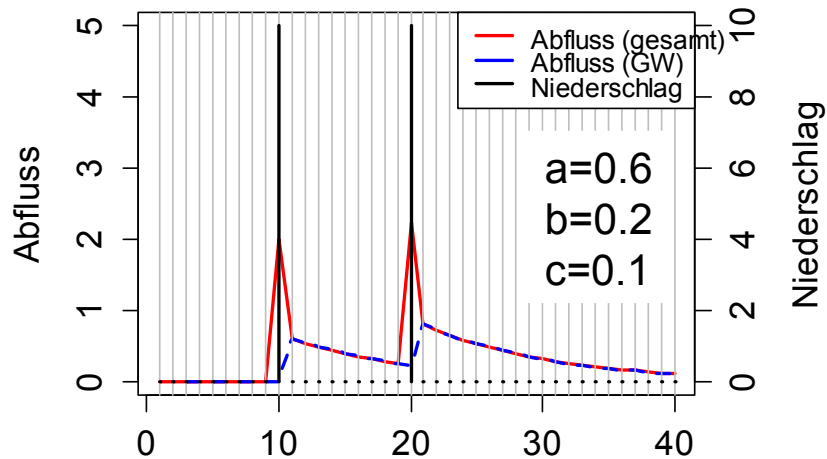
$$G_i = (1-c)G_{i-1} + aP_i$$

## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



Implementiere das *abc*-Modell als Funktion in R (*abc.R*) und bearbeite die weiteren Aufgaben, die in *abc.R* enthalten sind.

## Kontinuierlicher Wasserhaushalt mit dem *abc*-Modell



Zeitindex

Zeitindex



## Das abc-Modell auf dem Prüfstand

$$\frac{\Delta S}{\Delta t} = P - ET - Q$$

- ☒ Massenerhaltung als Grundprinzip
- ☐ Einfluss der Gebietsfeuchte auf Abflussbildung
- ☐ Abbildung physikalischer Prozesse (Verdunstung, Schnee)



Welche dieser Kriterien erfüllt das *abc*-Modell?

Weitere Schwächen des *abc*-Modells?