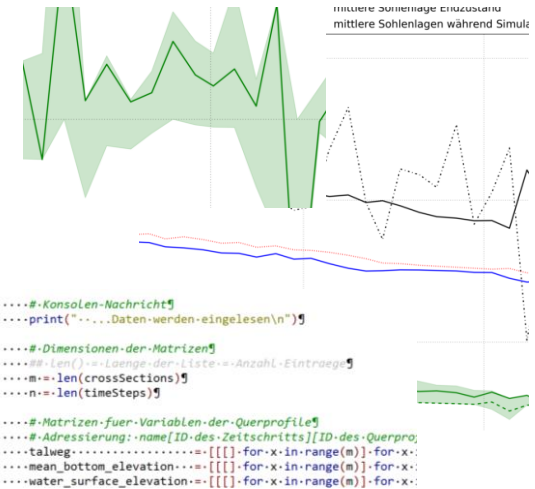
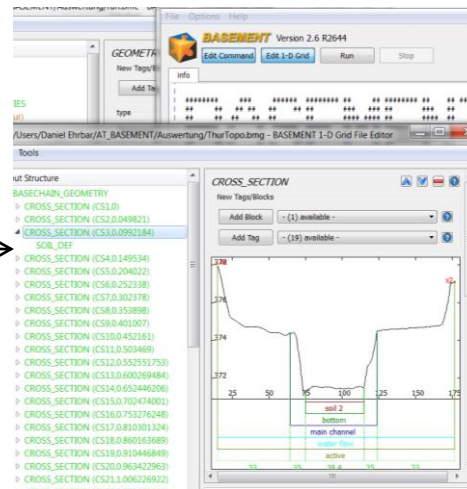
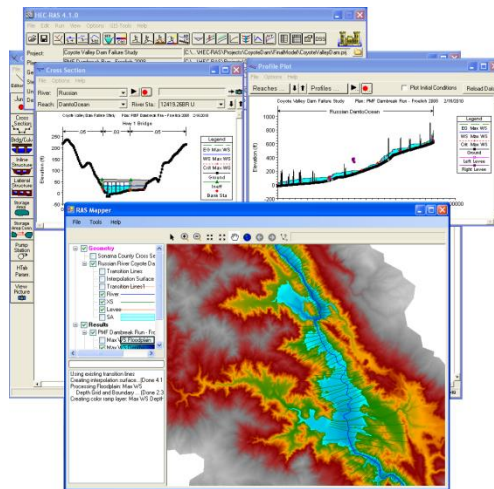


# 1D – BASEchain

## Pre- und Postprocessing



Daniel Ehrbar (ehrbar@vaw.baug.ethz.ch)  
Rapperswil, 27. Januar 2016



# Inhalt

- empfohlene Software
- Download-Links Software
- Input: Querprofil-Datei (\*.bmg)
- Pre-Processing: HECRAS2Basement
- Output: Standard-Resultatdatei (\*\_out.dat)
- Post-Processing: Skript-basierte Auswertung

N.B.: – Beispiele und Live-Demo mit Windows 7  
– Verwendung von BASEMENT v2.6 (unveröffentlicht)  
– Skript für Auswertung noch nicht online



## empfohlene Software

### Python:

- frei verfügbar
- Plattform-unabhängig
- einfach zu erlernende Sprache
- klare und übersichtliche Struktur
- als Skriptsprache geeignet
- zahlreiche nützliche Bibliotheken



### Editoren:

- Text-Editor: z.B. Notepad++
- Entwicklungsumgebung:  
z.B. PyScripter





## Download-Links Software



**Python:** z.B. Freemium-Distribution «Anaconda»

- beinhaltet Bibliotheken wie matplotlib 1.5.0, numpy 1.9.3, PyQt 4.11.4
- erhältlich für Windows, OS X und Linux
- Version: Python 2.7, 64-bit
- [https://www.continuum.io/downloads#\\_windows](https://www.continuum.io/downloads#_windows)



**Notepad++**

<https://notepad-plus-plus.org/download/v6.8.8.html>



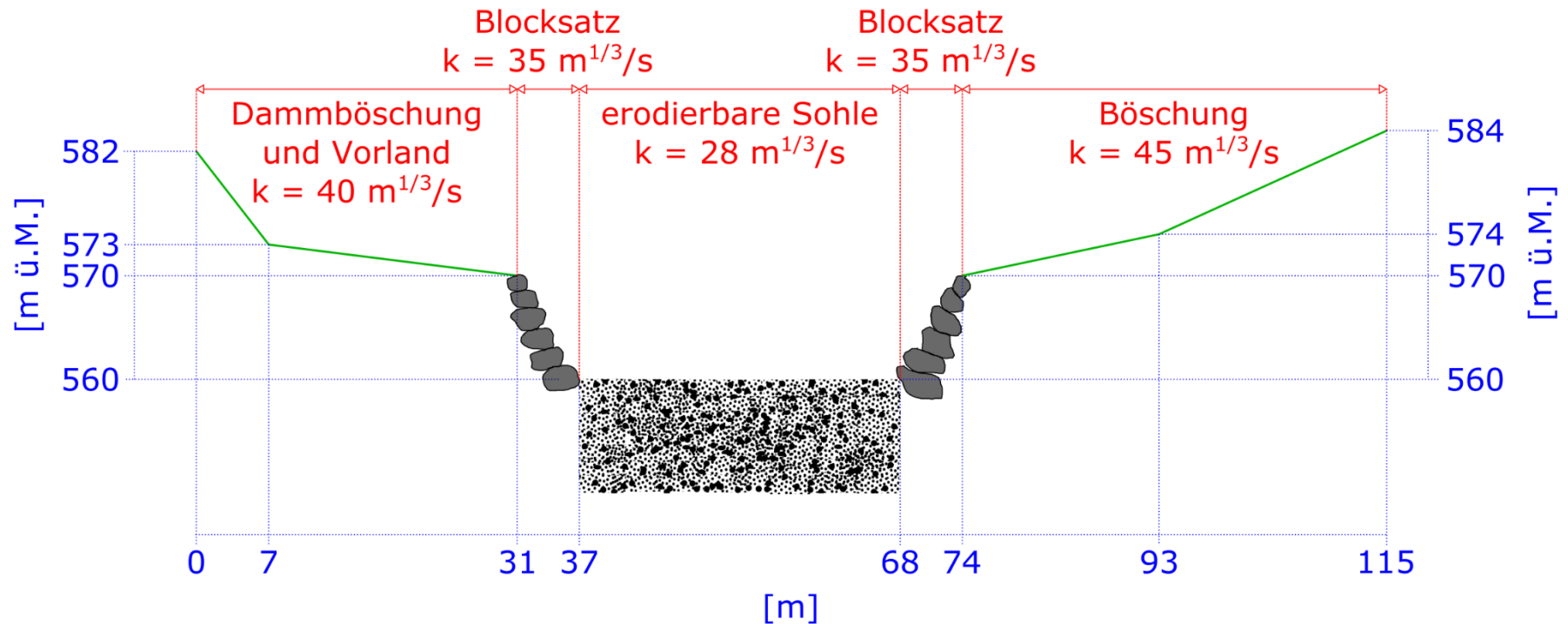
**PyScripter**

<http://sourceforge.net/projects/pyscripter>



# Input: Querprofil-Datei (\*.bmg)

(1/2)

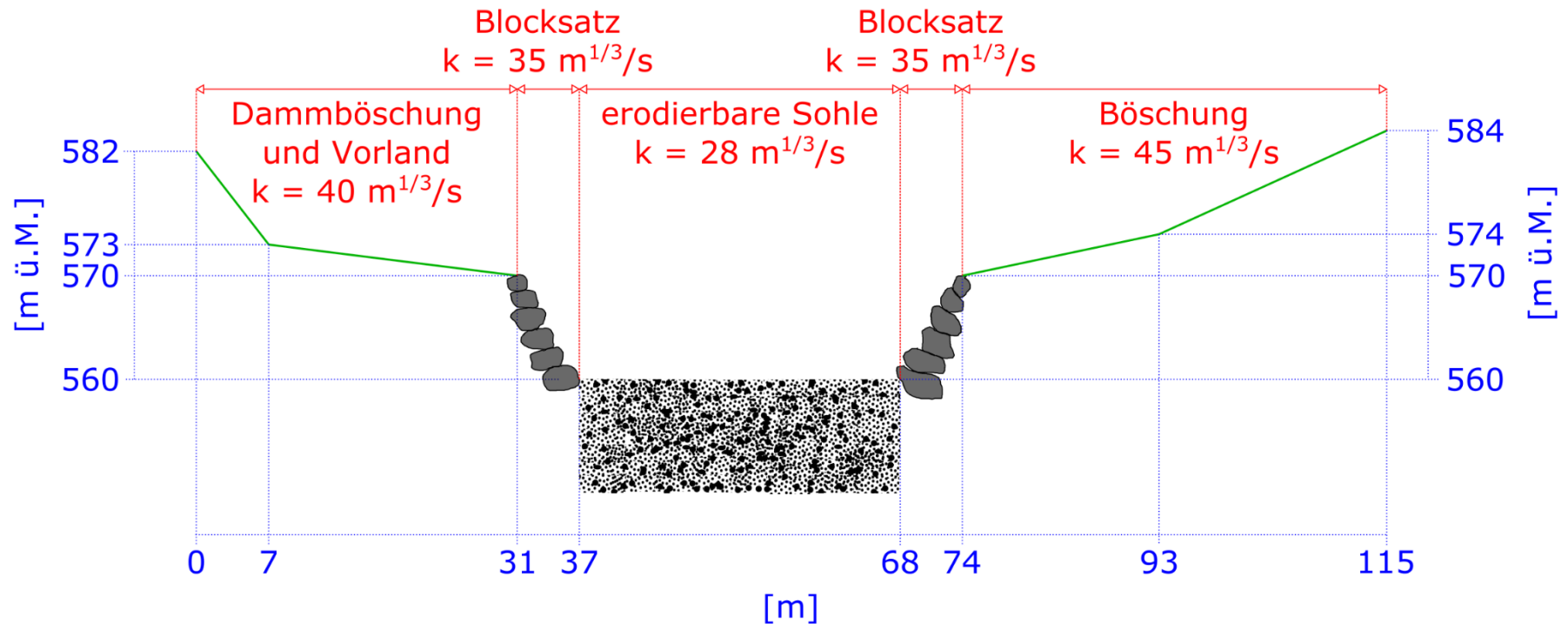


```
CROSS_SECTION {  
    name                = QP-7  
    distance_coord      = 0.0358  
    main_channel_range  = (31,74)  
    friction_ranges     = ( (0,31), (31,37), (37,68), (68,74), (74,115) )  
    friction_coefficients = (40, 35, 28, 35, 45)  
    node_coords         = ( (0,582), (7,573) ... (93,574), (115,584) )  
}
```



# Input: Querprofil-Datei (\*.bmg)

(2/2)



```
... // [Fortsetzung]
```

```
bottom_range = (37, 68)
```

```
SOIL_DEF {
```

```
    index = 1
```

```
    range = (37, 68)
```

```
}
```

```
}
```

} Bereich des Geschiebe-Transports



# Pre-Processing: HECRAS2Basement

(1/2)

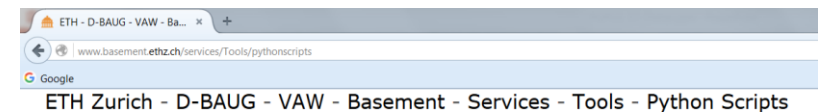
Ziel: Querprofile aus HEC-RAS in *BASEchain* importieren

Download:

- <http://www.basement.ethz.ch/services/Tools/pythonscripts>
- HECRAS2Basement.zip

5 Dateien:

- `crossSection.py`
- `crossSectionConversion.py`
- `HECRAS2basement.py`
- `mesh.py`
- `meshParser1D.py`



## Services

Software download

Tools

BASEmesh

BASEbreach

BREACH MACCHIONE

BASEGRAIN

DamBASE

**Python Scripts**

Tutorials

Test Cases

Workshops

Lectures

Users' Meetings

## Python Scripts

Transforms GEWISS cross section data to the bmg-format: [GEWISS2Basement.py](#)

Transforms Floris Input files for 1D to the new bmg-format: [Funin2Basement.zip](#) (6 KB)

Transforms HEC-RAS Input files for 1D to the new bmg-format: [HECRAS2Basement.zip](#) (19KB)



# Pre-Processing: HECRAS2Basement

(2/2)

Ausführung über Eingabeaufforderung (cmd):

```
python HECRAS2basement.py <Querprofildatei.g01>
```

konvertierte Querprofile:

```
converted.bmg
```

Reihenfolge der Querprofile:

```
converted_cross_section_names.txt
```

**→ LIVE-DEMO**

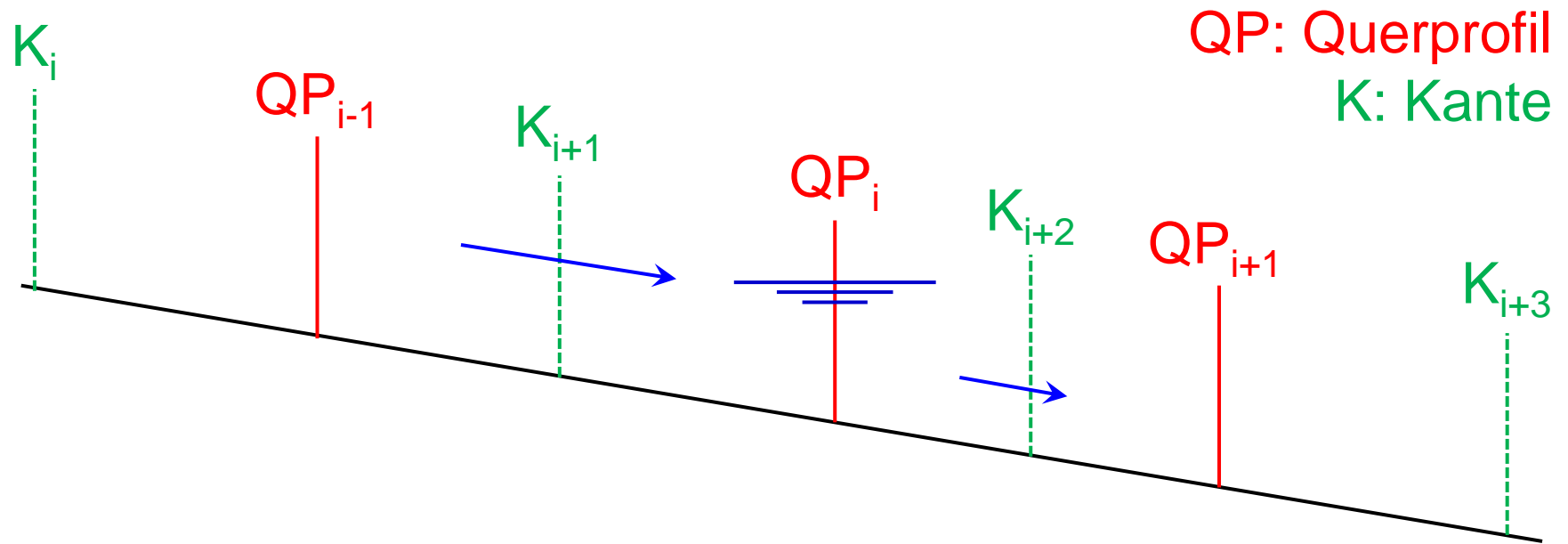
Dokumentation:

```
BASEchain_HEC-RAS-Querprofile_importieren.pdf
```





## Output: Standard-Resultatdatei (\*\_out.dat)



Inhalt: für Initialzustand ( $t = 0$ ) und jeden Output-Zeitschritt...  
...Zustände in den Querprofilen (z.B. Wasserspiegel in m ü.M.)  
...Massenflüsse über Kanten (z.B. Abfluss in  $\text{m}^3/\text{s}$ )

- Text-Datei mit Bezeichnung `<region_name>_out.dat`
- wird unabhängig von `SPECIAL_OUTPUT` *immer* geschrieben



# Post-Processing: Skript-basierte Auswertung

(1/3)

Ziel: Abbildungen aus Standard-Resultatdatei generieren

→ Verwendung eines Python-Skripts der HSR

Download:

- <http://www.basement.ethz.ch/services/Tools/pythonscripts>
- `Auswertung_1d_HSR_stationaer.py` und/oder `Auswertung_1d_HSR.py`

2 Dateien:

- `Auswertung_1d_HSR_stationaer.py` für stationäre rein hydraulische Simulationen
- `Auswertung_1d_HSR.py` für allgemeine Simulationen



# Post-Processing: Skript-basierte Auswertung

(2/3)

Skript	stationäre Version	allgemeine Version
Anwendung	stationäre, rein hydraulische Simulationen (feste Sohle)	instationäre Simulationen, auch mit Sedimenttransport (bewegliche Sohle)
Diagramme	<b>Längenprofil</b> <ul style="list-style-type: none"><li>• Dammoberkanten</li><li>• Energielinienlage</li><li>• Wasserspiegellage</li><li>• mittlere Sohlenlage *</li></ul> <b>Froude-Diagramm</b>	<b>Längenprofil</b> <ul style="list-style-type: none"><li>• Dammoberkanten</li><li>• maximale Energielinienlage</li><li>• maximale Wasserspiegellage</li><li>• mittlere Sohlenlagen *</li></ul> <b>Froude-Diagramm</b> <b>Sohlschubspannungs-Diagramm</b> <b>Transport-Diagramm</b> <ul style="list-style-type: none"><li>• gesamte Geschiebe-Fracht</li><li>• gesamte Schwebstoff-Fracht</li></ul> <b>Differenzen-Diagramm</b> <ul style="list-style-type: none"><li>• Auflandungsmächtigkeiten</li><li>• Erosionstiefen</li><li>• resultierende Sohlenveränderung</li></ul>

\* falls kein `bottom_range` definiert ist, wird die mittlere Höhenlage des Querprofils ausgegeben



# Post-Processing: Skript-basierte Auswertung

(3/3)

automatische Ausführung bei Doppelklick

→ **LIVE-DEMO**

individuelle *Anpassungen* des Skripts möglich, z.B.

- Inhalt: Daten hinzufügen oder entfernen
- Form: Schriftgrösse und –art, Linienstärke und –farben etc.

→ **LIVE-DEMO**

Dokumentation:

`BASEchain_Skript-basierte_Datenauswertung.pdf`