

# Projektmanagement (Softwarepraktikum)

**Thema: Workflows**

# Typisches Szenario in der Praxis

**Benötigt:** Auswertung biologischer Massendaten z.B.

- NGS
- Massenspektrometrie
- Mikroskopie

# Typisches Szenario in der Praxis

**Benötigt:** Auswertung biologischer Massendaten z.B.

- NGS
- Massenspektrometrie
- Mikroskopie

**(Leider) selten!**

Es existiert ein Tool für  
die gesamte Analyse!



# Typisches Szenario in der Praxis

**Benötigt:** Auswertung biologischer Massendaten z.B.

- NGS
- Massenspektrometrie
- Mikroskopie

**(Leider) selten!**

Es existiert ein Tool für die gesamte Analyse!



**(Zum Glück) selten!**

Es existiert noch gar keine Software! Ich muss alles selbst entwickeln!



# Typisches Szenario in der Praxis

**Benötigt:** Auswertung biologischer Massendaten z.B.

- NGS
- Massenspektrometrie
- Mikroskopie

**(Leider) selten!**

Es existiert ein Tool für die gesamte Analyse!



**(Zum Glück) selten!**

Es existiert noch gar keine Software! Ich muss alles selbst entwickeln!



**Häufig!**

Es existieren Tools für einzelne Schritte der Analyse!



# Typisches Szenario in der Praxis

**Benötigt:** Auswertung biologischer Massendaten z.B.

- NGS
- Massenspektrometrie
- Mikroskopie

**(Leider) selten!**

Es existiert ein Tool für die gesamte Analyse!



**(Zum Glück) selten!**

Es existiert noch gar keine Software! Ich muss alles selbst entwickeln!

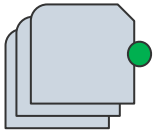


**Häufig!**

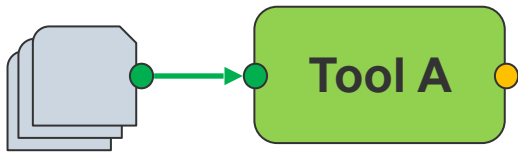
Es existieren Tools für einzelne Schritte der Analyse!



# Einzelne Tools → Analyse Pipeline

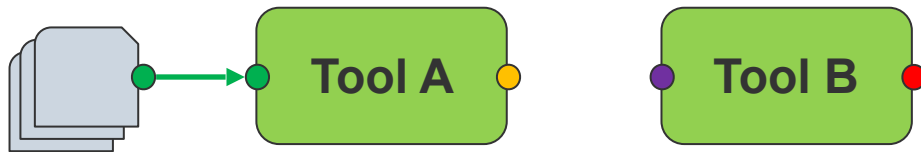


# Einzelne Tools → Analyse Pipeline

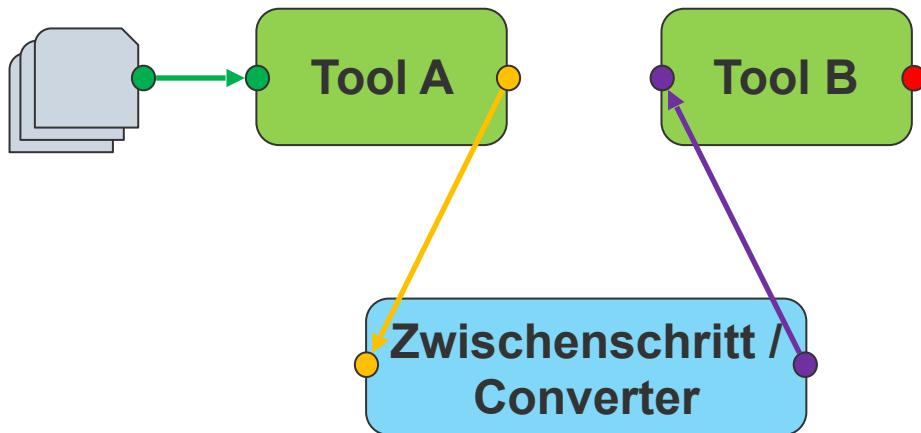




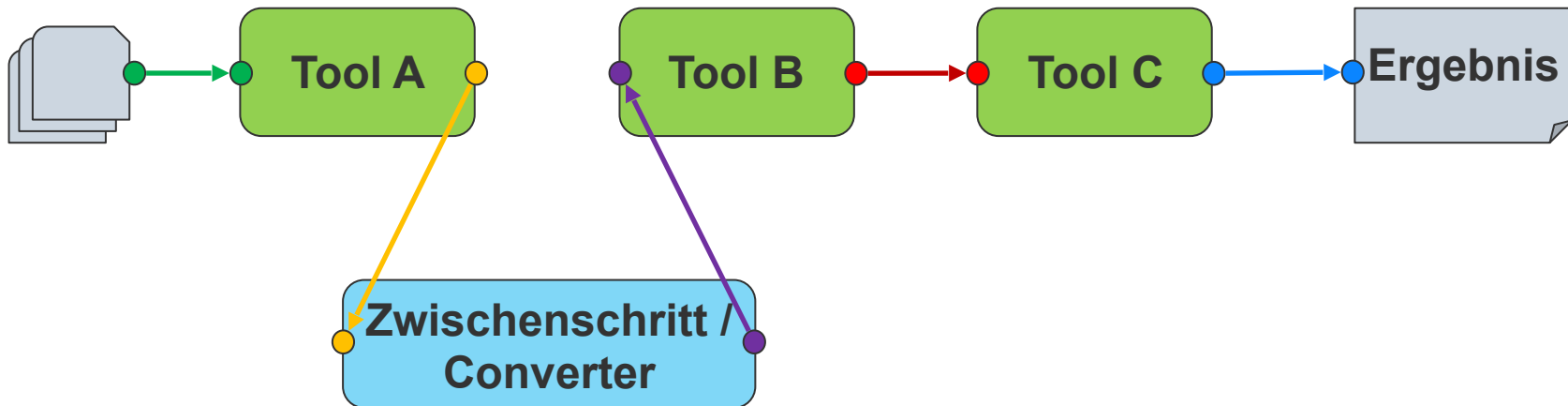
# Einzelne Tools → Analyse Pipeline



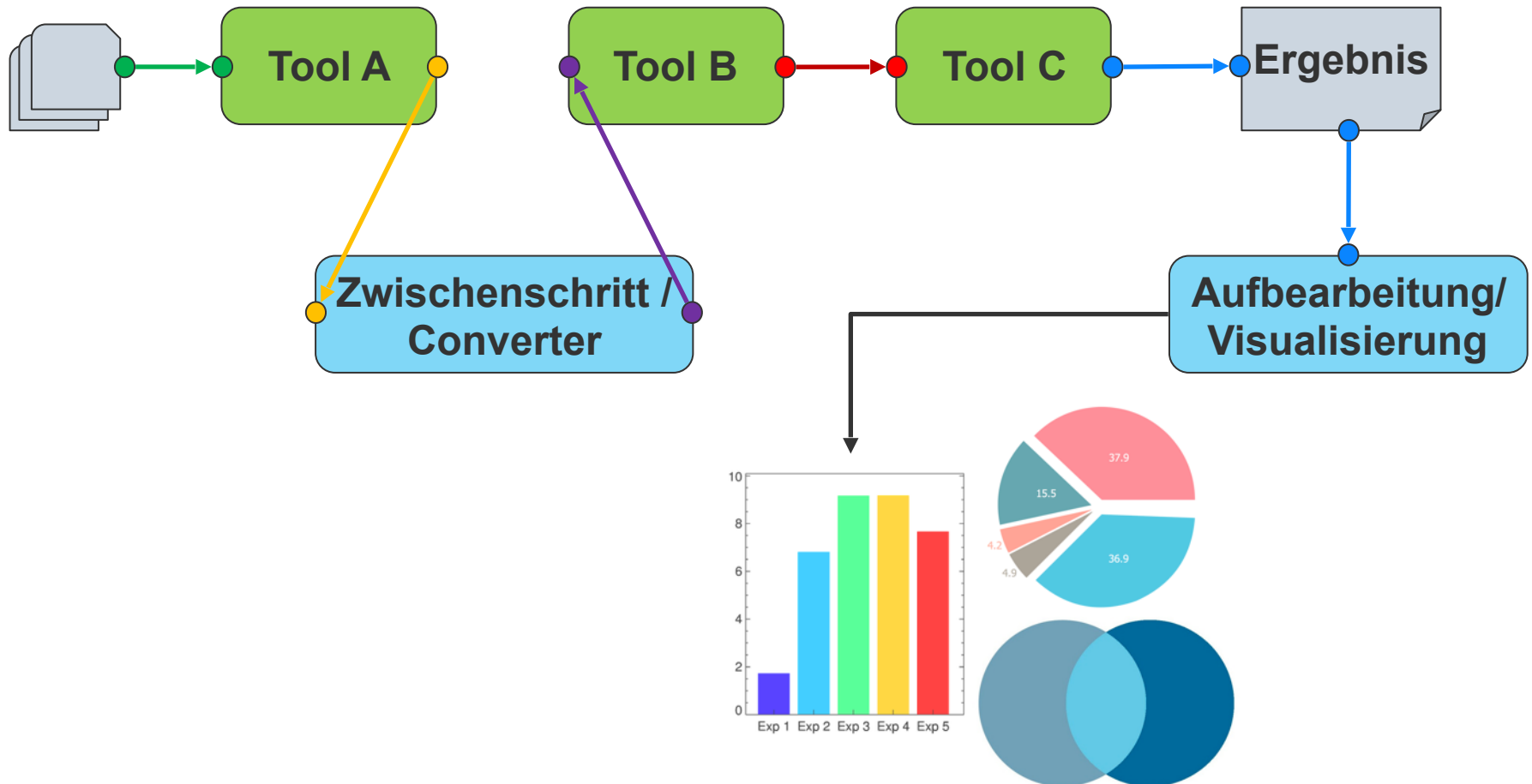
# Einzelne Tools → Analyse Pipeline



# Einzelne Tools → Analyse Pipeline



# Einzelne Tools → Analyse Pipeline



# Realisierung

1. Von Hand Tools nacheinander ausführen

# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...

# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines

# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines
  - Script basiert:



# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines
  - Script basiert:
    - Snakemake

```
rule targets:
    input:
        "plots/dataset1.pdf",
        "plots/dataset2.pdf"

rule plot:
    input:
        "raw/{dataset}.csv"
    output:
        "plots/{dataset}.pdf"
    shell:
        "somecommand {input} {output}"
```

# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines
  - Script basiert:
    - Snakemake
    - Nextflow

```
#!/usr/bin/env nextflow

params.in = "$baseDir/data/sample.fa"
sequences = file(params.in)

/*
 * split a fasta file in multiple files
 */
process splitSequences {

    input:
    file 'input.fa' from sequences

    output:
    file 'seq_*' into records

    """
    awk '/^>/{f="seq_"++d} {print > f}' < input.fa
    """

}

/*
 * Simple reverse the sequences
 */
process reverse {

    input:
    file x from records

    output:
    stdout result

    """
    cat $x | rev
    """

}
```

# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines
  - Script basiert:
    - Snakemake
    - Nextflow
  - GUI basiert:

```
#!/usr/bin/env nextflow

params.in = "$baseDir/data/sample.fa"
sequences = file(params.in)

/*
 * split a fasta file in multiple files
 */
process splitSequences {

    input:
    file 'input.fa' from sequences

    output:
    file 'seq_*' into records

    """
    awk '/^>/{f="seq_"+d} {print > f}' < input.fa
    """

}

/*
 * Simple reverse the sequences
 */
process reverse {

    input:
    file x from records

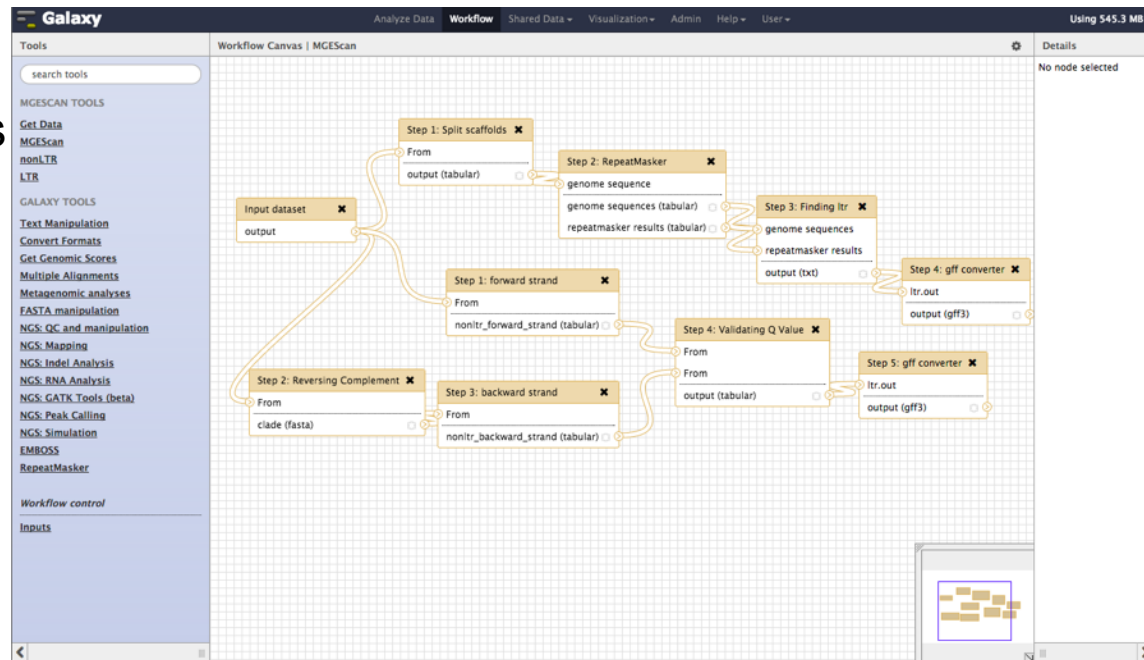
    output:
    stdout result

    """
    cat $x | rev
    """

}
```

# Realisierung

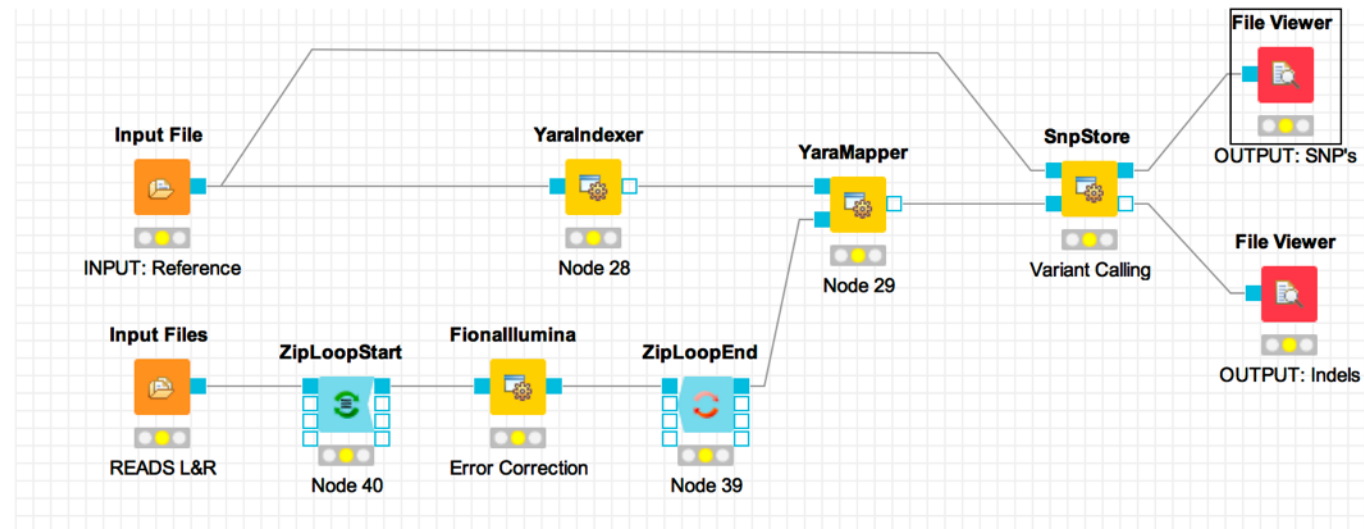
1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines
  - Script basiert:
    - Snakemake
    - Nextflow
  - GUI basiert:
    - Galaxy



# Realisierung

1. Von Hand Tools nacheinander ausführen
2. Eigenes “framework”
  - Batch script
  - Python script
  - ...
3. Generische workflow engines

- Script basiert:
  - Snakemake
  - Nextflow
- GUI basiert:
  - Galaxy
  - KNIME



# Ablauf des Praktikums

1. Tutorials zur Einarbeitung in KNIME und Snakemake
2. Literaturrecherche zur Auswahl einer aktuellen, relevanten und relativ komplexeren Analyse
3. Weitergehende Recherche nach publizierten '*state of the art*' Workflows
4. Nachbau, Erweiterung, Verbesserung der Workflows in KNIME und Snakemake
5. Evaluierung auf realen Daten

# Ablauf des Praktikums

1. Tutorials zur Einarbeitung in KNIME und Snakemake
2. Literaturrecherche zur Auswahl einer aktuellen, relevanten und relativ komplexeren Analyse
3. Weitergehende Recherche nach publizierten '*state of the art*' Workflows
4. Nachbau, Erweiterung, Verbesserung der Workflows in KNIME und Snakemake
5. Evaluierung auf realen Daten

## **Quantitative Aufteilung: (in %)**

Praktische Programmierarbeit: 50%

Soft Skills: 50%

## **Verwendete Programmiersprache(n):**

R, Python oder andere Skriptsprache

## **Schwierigkeitsgrad**

A Programmieren \*\*\*\*

B Biologie/Chemie \*

C Projektmanagement \*\*\*

**Erforderliche Vorkenntnisse:** R, Python

# Ablauf des Praktikums

1. Tutorials zur Einarbeitung in KNIME und Snakemake
2. Literaturrecherche zur Auswahl einer aktuellen, relevanten und relativ komplexeren Analyse
3. Weitergehende Recherche nach publizierten ‘*state of the art*’ Workflows
4. Nachbau, Erweiterung, Verbesserung der Workflows in KNIME und Snakemake
5. Evaluierung auf realen Daten

Zeitplan	
Vorbesprechung	1.3.
Tutorials	19.3. – 23.3.
Recherche zu Analyse und Workflows	24.3. – 4.4.
Bearbeitung	bis 17.5.

## Quantitative Aufteilung: (in %)

Praktische Programmierarbeit: 50%

Soft Skills: 50%

## Verwendete Programmiersprache(n):

R, Python oder andere Skriptsprache

## Schwierigkeitsgrad

A Programmieren \*\*\*\*

B Biologie/Chemie \*

C Projektmanagement \*\*\*

**Erforderliche Vorkenntnisse:** R, Python