

# Titel der Präsentation bearbeiten

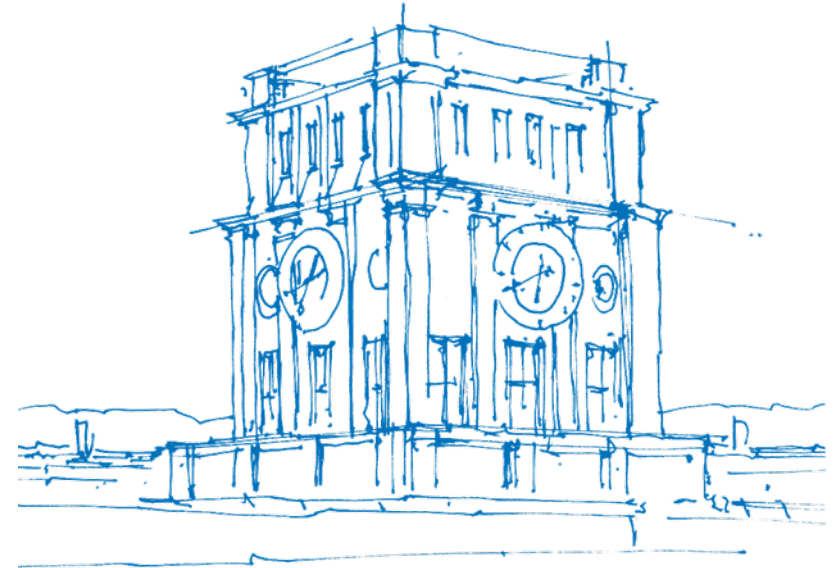
Alex Hocks   Jan Hampe   Johannes Riemenschneider

Technische Universität München

@Fakultät@

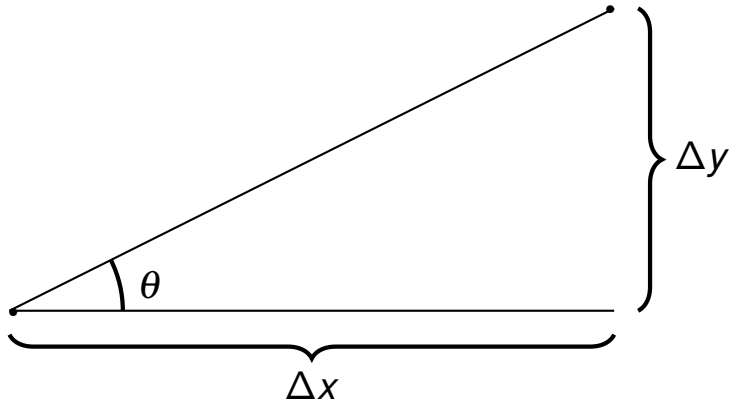
@LehrstuhlName@

3. November 2022



*TUM Uhrenturm*

## 2 Dimensional Force Calculation



$$\Delta x = x_2 - x_1 \quad (1)$$

$$\Delta y = y_2 - y_1 \quad (2)$$

$$|F| = \frac{m_1 m_2}{\Delta x^2 + \Delta y^2} \quad (3)$$

$$F_x = \cos(\theta) \cdot |F| = \Delta x \cdot \frac{m_1 m_2}{(\Delta x^2 + \Delta y^2)^{3/2}} \quad (4)$$

$$F_y = \sin(\theta) \cdot |F| = \Delta y \cdot \frac{m_1 m_2}{(\Delta x^2 + \Delta y^2)^{3/2}} \quad (5)$$

# Utilizing $F_{ij} = -F_{ji}$

The naive approach ( $n \cdot (n - 1)$  Force calculations):

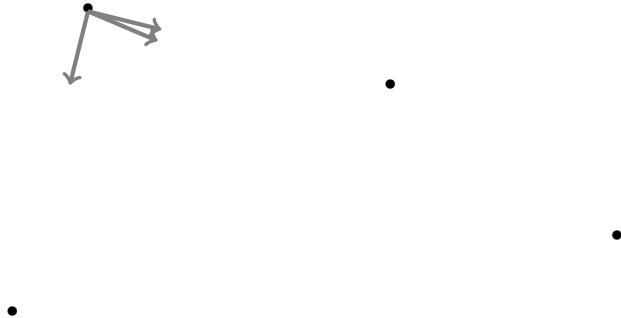
```
for all Particles p:
    for all Particles p'!=p:
        computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

The naive approach ( $n \cdot (n - 1)$  Force calculations):

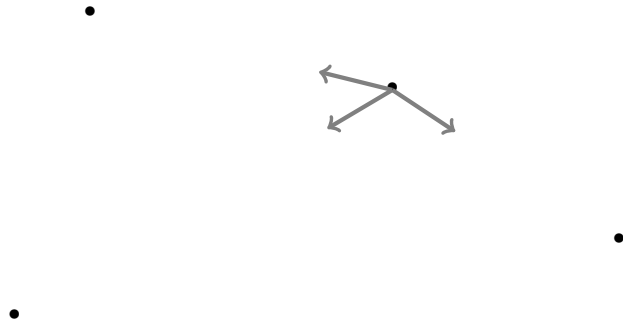
```
for all Particles p:
    for all Particles p'!=p:
        computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

The naive approach ( $n \cdot (n - 1)$  Force calculations):

```
for all Particles p:
    for all Particles p'!=p:
        computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

The naive approach ( $n \cdot (n - 1)$  Force calculations):

```
for all Particles p:
    for all Particles p'!=p:
        computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

A better approach ( $\frac{1}{2}n \cdot (n-1)$  Force calculations):

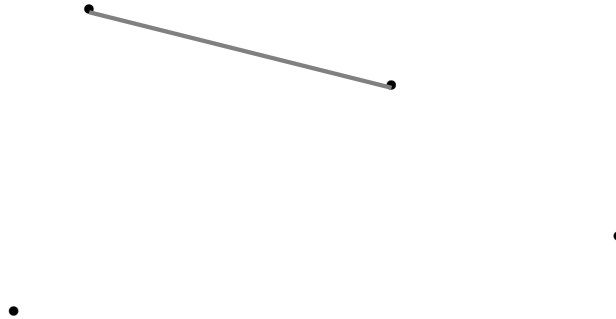
```
for all ParticlePairs (p,p'):
    computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

A better approach ( $\frac{1}{2}n \cdot (n-1)$  Force calculations):

```
for all ParticlePairs (p,p'):
    computeF(p,p')
```

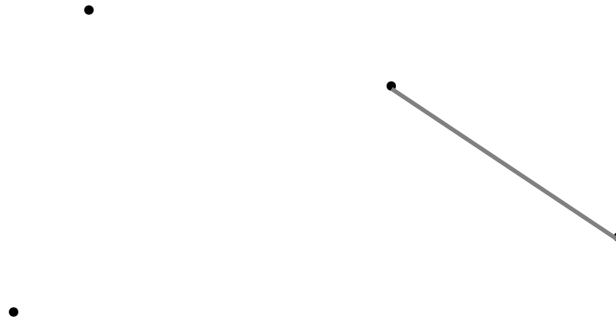




# Utilizing $F_{ij} = -F_{ji}$

A better approach ( $\frac{1}{2}n \cdot (n-1)$  Force calculations):

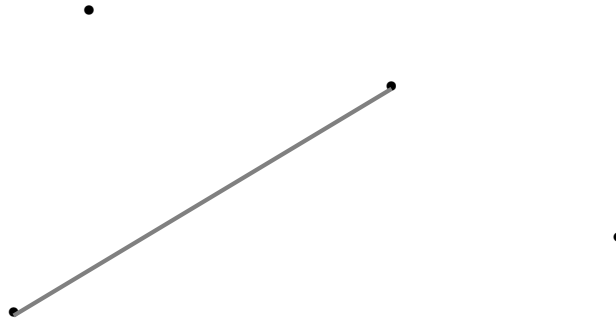
```
for all ParticlePairs (p,p'):
    computeF(p,p')
```



# Utilizing $F_{ij} = -F_{ji}$

A better approach ( $\frac{1}{2}n \cdot (n - 1)$  Force calculations):

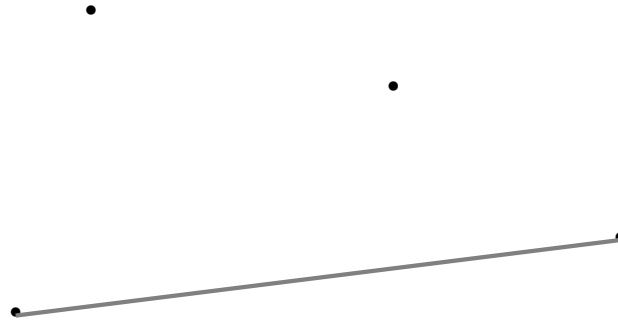
```
for all ParticlePairs (p,p'):
    computeF(p,p')
```



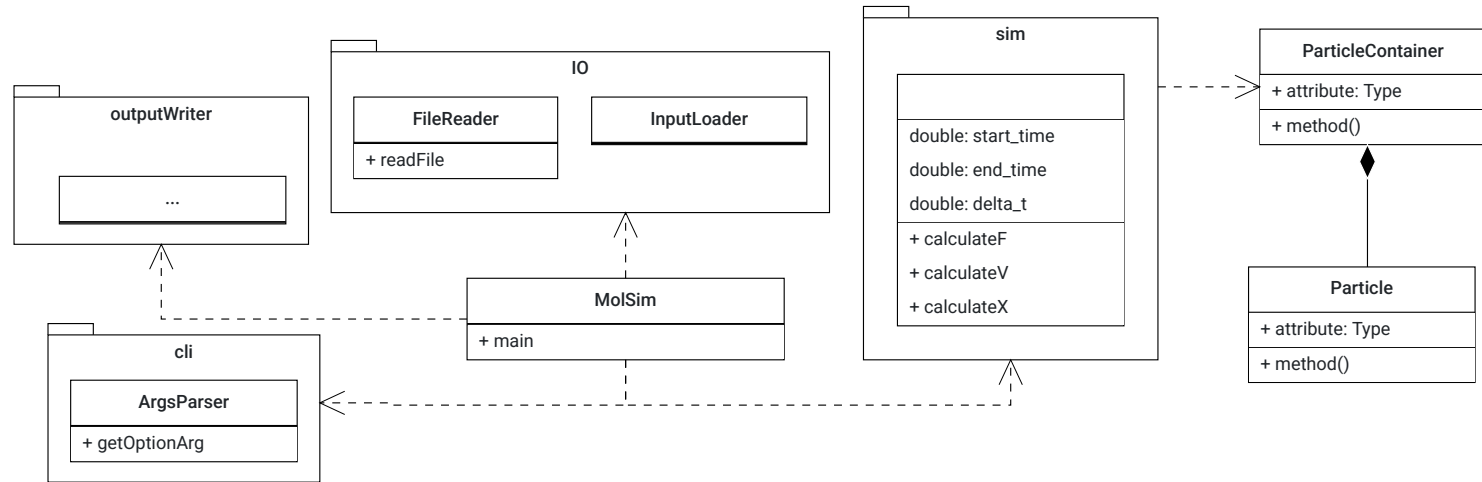
# Utilizing $F_{ij} = -F_{ji}$

A better approach ( $\frac{1}{2}n \cdot (n - 1)$  Force calculations):

```
for all ParticlePairs (p,p'):
    computeF(p,p')
```



# Refactoring



# Use of the Program

```
./MolSim --help
```

Welcome to MolSim Help

Usage:

```
MolSim <input-files> <options>
```

Options:

<code>--help, -h</code>	Prints this screen.
<code>-dt &lt;value&gt;</code>	Sets delta time to <value>. If -dt is not specified default value is used.
<code>-et &lt;value&gt;</code>	Set end time to <value>. If -et is not specified default value is used.
<code>-o &lt;name&gt;</code>	Set base name of output files. DO NOT USE A PATH! Default is 'result'.
<code>-of &lt;path&gt;</code>	Set path to output folder. Default is ./output

# Generic IO

```
template <typename LOCATOR, void (*LOAD)(LOCATOR, std::list<Particle>&)>
class InputLoader {
    private:
        std::list<Particle> buffer;
        LOCATOR locator;
    public:
        explicit InputLoader(LOCATOR loc) : locator(loc) {}
        InputLoader(const InputLoader& i) = delete;
        void reload() { LOAD(locator, buffer); }
        void getParticles(std::vector<Particle> &buf) {...}
};
```

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same



# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

What happened:

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

What happened:

- floating point operations are not associative → both outputs were „correct“

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

What happened:

- floating point operations are not associative → both outputs were „correct“
- probably some compiler magic

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

What happened:

- floating point operations are not associative → both outputs were „correct“
- probably some compiler magic
- Funfact: Compiling the exact same code with the exact same compiler settings will result in the same output

# Refactoring surprises

- Task: Verify the correctness of the program after major refactoring
- Idea: string-compare the new output with the old output. If the program works as intended it should be the same
- Observation: Very slight differences in the outputs generated (unoberservable in the videos)

What happened:

- floating point operations are not associative → both outputs were „correct“
- probably some compiler magic
- Funfact: Compiling the exact same code with the exact same compiler settings will result in the same output
- First-hand encounter of the inaccuracies of numerical programming

# Grundlage der Masterfolien

Als Grundlage dient der Corporate Design Style Guide der TUM.

Die Präsentationsvorlage ist auf gute Lesbarkeit und klare Darstellung von Informationen optimiert.

# Hier steht eine 2-zeilige Überschrift

Als Grundlage dient der Corporate Design Style Guide der TUM.

Die Präsentationsvorlage ist auf gute Lesbarkeit und klare Darstellung von Informationen optimiert.



# Schrift

Das Grundprinzip ist, Informationen bestmöglich zu transportieren. Dazu muss vor allem die Schrift einheitlich und für alle im Raum lesbar sein.

Schriftart: Helvetica

Schriftgrößen: 25 | 18 | 14 | 11

Zeilenabstand: 1,15 mm

Die Einstellungen sind für diese Vorlage als Standard eingestellt. Bei Diagrammen und Tabellen muss die Schriftgröße ggf. angepasst werden. Für Auszeichnungen im Fließtext kann auch **fett** markiert werden. Bei großer Distanz bzw. kleinem Präsentationsmedium kann der Schriftgrad notfalls proportional erhöht werden.

# Farben

Als erstes soll mit schwarz und weiß gearbeitet werden.

Für Aufwändigere Darstellungen sind Farben mit Bedacht und in möglichst geringem Umfang einzusetzen.

In diesem Folienmaster ist die Farbpalette festgelegt.

Zuerst mit den Primärfarben arbeiten.



Für z.B. komplexe Diagramme stehen noch Sekundärfarben zur Verfügung.



Bei weiterer Komplexität oder zusätzlichen Markierungen:



# Texte

Kurze und knappe Texte, Fließtexte linksbündig, kein Blocksatz

Beispiel:

Tem soluptam, nisi as verum ereprehendam at acculpa quidisq uissit volupta tusdant utem as etur, odi odis es doluptiae dem nimaion con nossinctenis pora quam voloria consenimus blabore everfer epeliquo maio etur.

# Aufzählung

Bei kleinen Aufzählungen auf Aufzählungszeichen verzichten und ggf. zusätzliche Leerzeile.  
Nur die wesentlichen Punkte nennen und Themen auf verschiedene Seiten splitten.

Punkt 1

Punkt 2

Wenn Unterpunkte in einer Aufzählung nötig sind ist ein Einrücken mit – möglich

- Unterpunkt 1
  - Unterpunkt 1
  - Unterpunkt 2

Bei größeren Listen die Standardeinstellung • verwenden

- Unterpunkt 1
- Unterpunkt 2
- Unterpunkt 3

# Bilder – Allgemein

schlichte Darstellung von Informationen

reduzierte Farben

Rahmen und Überlagerungen nach Möglichkeit vermeiden

# Bilder

Bildbeschreibung

oberer Bildrand: Begrenzung durch Text



# Bilder

## Überschrift 2

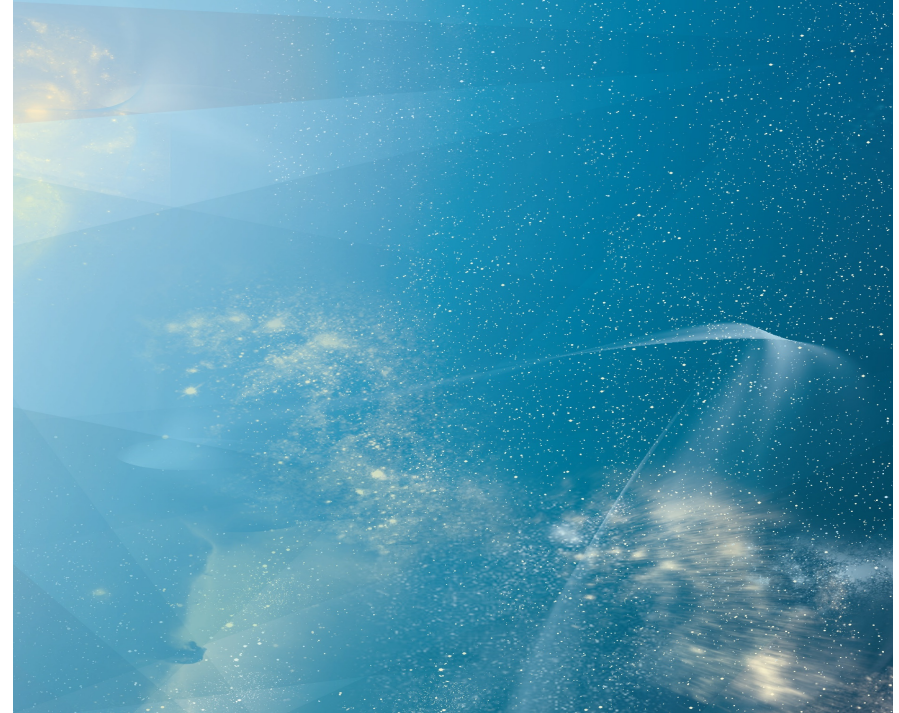
Hier steht ein einleitender oder beschreibender Fließtext und nach Wunsch eine Aufzählung.

Punkt 1

Punkt 2

Punkt 3

Punkt 4



# Bilder

Bildbeschreibung

oberer Bildrand: Begrenzung durch Text

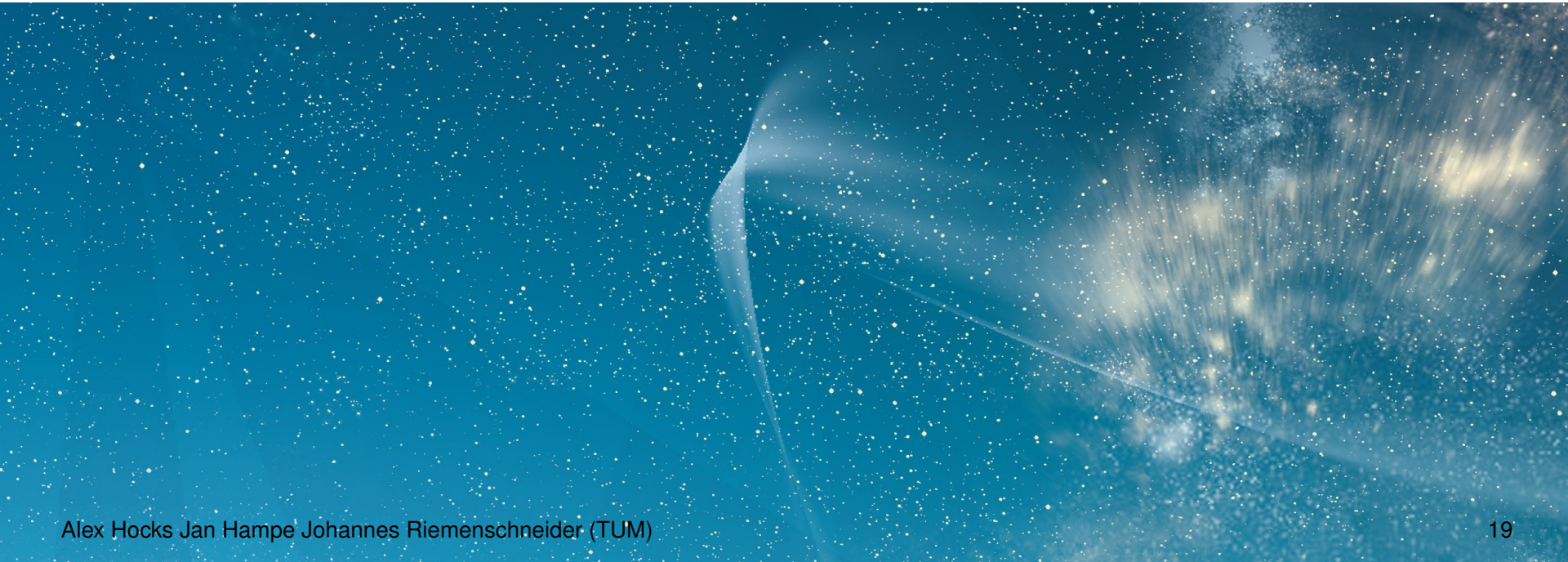




# Bilder

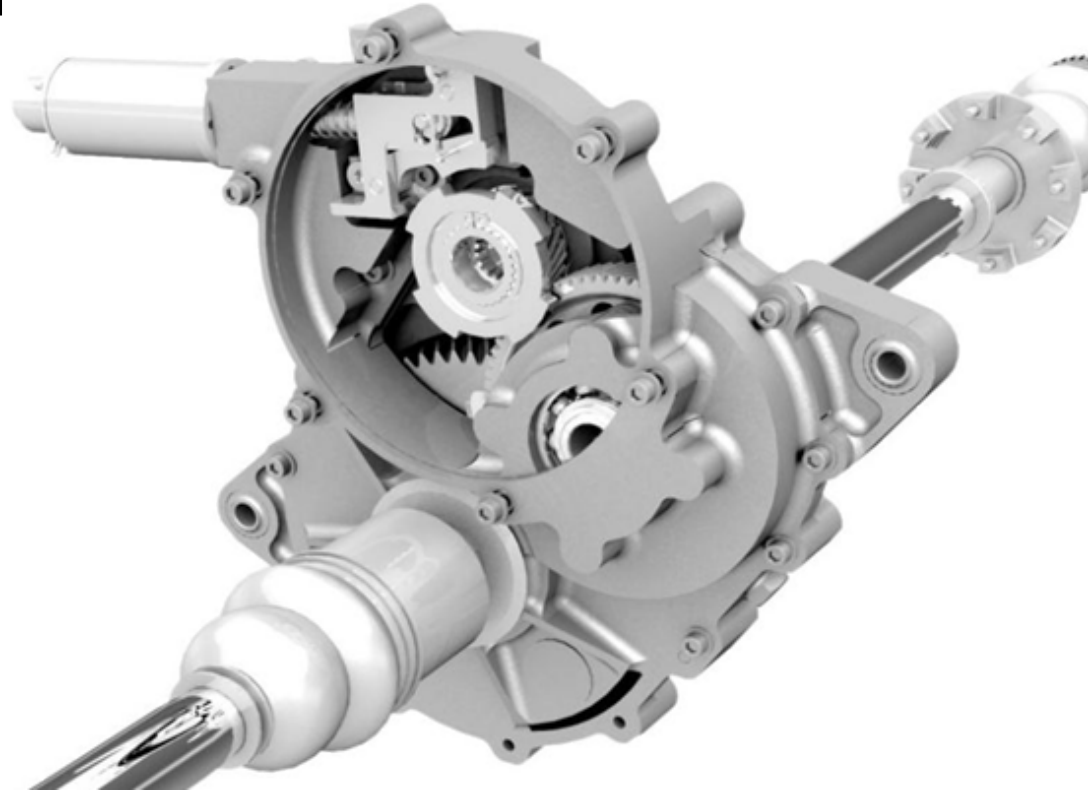
Bildbeschreibung

oberer Bildrand: Begrenzung durch Text



# Nicht Format füllende Bilder

Weißer bzw. transparenter Hintergrund  
mit genug Freiraum anordnen





# Bilder Format füllend – maximale Bildgröße



# Nicht Format füllende Bilder

Alternativ mit formatfüllendem Hintergrund: Weiß 5% dunkler  
Beschriftungen können zusätzlich neben den Bildern angebracht werden

Bilderklärung



# Tabelle – Beispiel 1

Tabelle ohne Farbe und kein Rand

innerer Seitenrand links 0 cm, um Faktor 1,75 skalierte Tabelle (für genug Zeilenabstand)

Ø - Strecke	39 km/Tag (14.360 km/Jahr)
Ø - Geschwindigkeit	25 km/h
Ø - Verfügbare Ladezeit	22 h/Tag
Kosten	Kleinwagen mit Verbrennungsmotor
Einsatzgebiet	Stadt und Umland

# Tabelle – Beispiel 2

Tabelle ohne Farbe und mit Rand

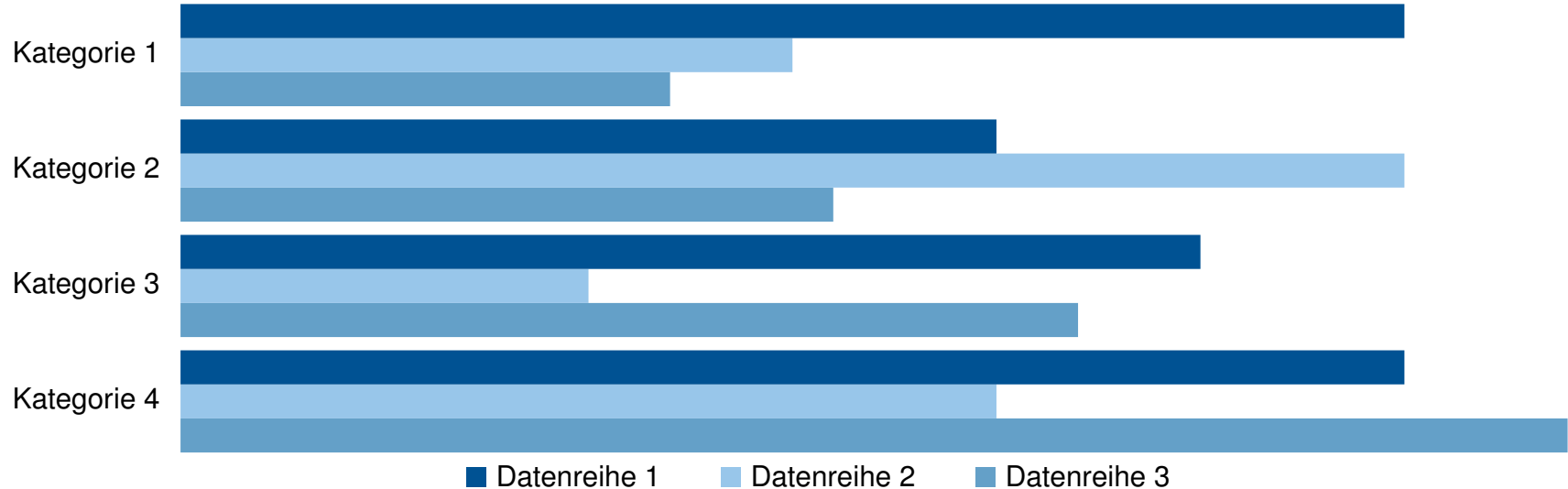
automatische Zelleninnenabstände, um Faktor 1,75 skalierte Tabelle (für genug Zeilenabstand)

Ø - Strecke	39 km/Tag (14.360 km/Jahr)
Ø - Geschwindigkeit	25 km/h
Ø - Verfügbare Ladezeit	22 h/Tag
Kosten	Kleinwagen mit Verbrennungsmotor
Einsatzgebiet	Stadt und Umland

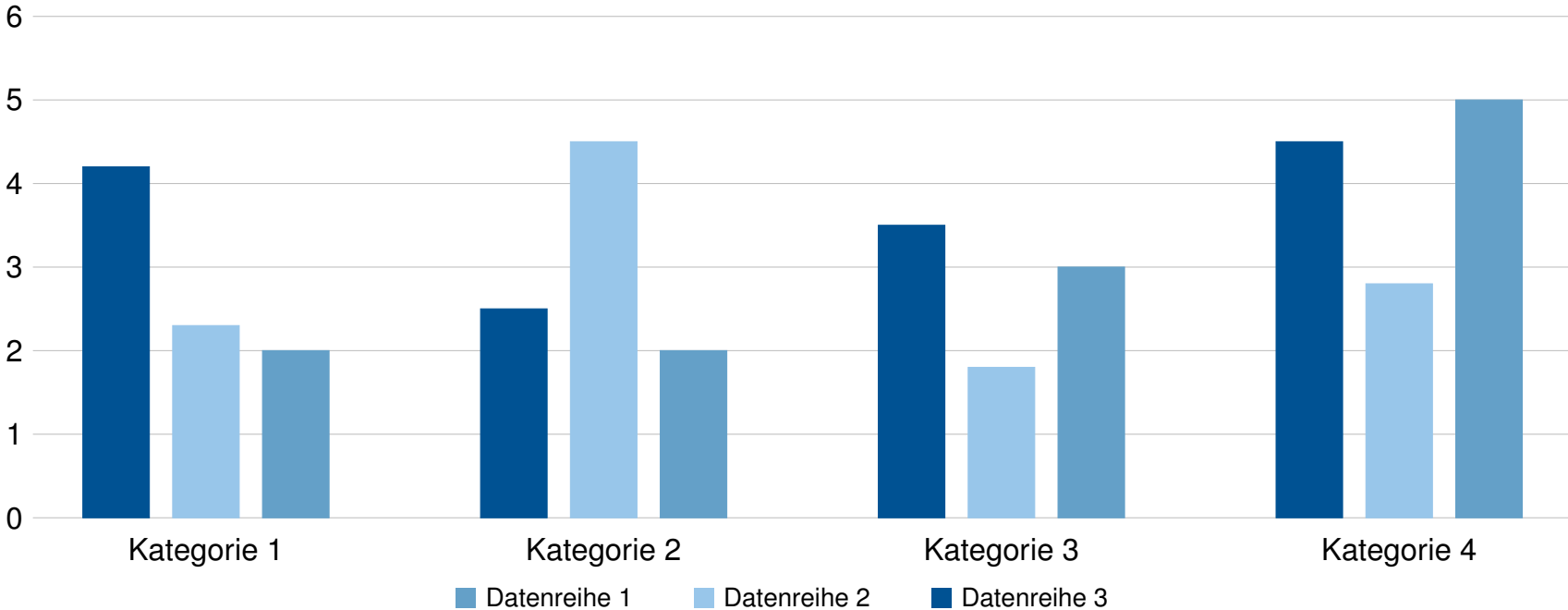
# Diagramme – Beispiel 1

Nach Möglichkeit linksbündig bleiben

Unnötige Striche und Balken vermeiden



# Diagramme





# Titel der Präsentation bearbeiten

Alex Hocks   Jan Hampe   Johannes Riemenschneider

Technische Universität München

@Fakultät@

@Lehrstuhlname@

3. November 2022



TUM Uhrenturm

Alex Hocks   Jan Hampe   Johannes Riemenschneider

Technische Universität München

@Fakultät@

@LehrstuhlName@

3. November 2022



# Präsentationsmuster

## kann auch als Kapiteltrenner verwendet werden