

# Informatik an der Universität Bern

## Einführung für neu eintretende Studierende

**Prof. Dr. Timo Kehrer**

*u*<sup>b</sup>

# Übersicht

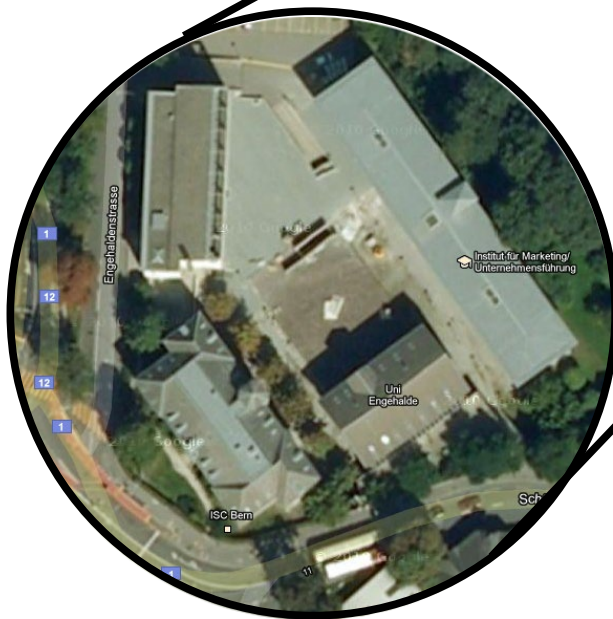
- Institut für Informatik (INF)
  - Lageplan
  - Personen
  - Forschung
- Bachelorstudium Informatik
  - Aufbau des Studiums
  - Vorlesungen
  - Prüfungen

*u<sup>b</sup>*

# Institut für Informatik

# Lageplan

- **INF Uni Engehälde**
  - Neubrückstr. 10
  - teils Schützenmattstr. 14
  - teils Engehaldenstr. 8





# $u^b$ Areal Uni Engehalde



- Neubrückstrasse 10
- Schützenmattstrasse 14



- Engehaldenstrasse 8

*u<sup>b</sup>*

# Personen am Institut für Informatik

- ca. 10 Professoren/innen und Dozenten/innen
- 3 Sekretariatsleiterinnen
- 1 Systemadministrator
- ca. 70 wissenschaftliche Mitarbeiter/innen
- ca. 450 Studierende (Haupt- und Nebenfach Informatik)

# Studierende





$u^b$

# Professoren/innen und Dozenten



David Bommes



Torsten Braun



Christian Cachin



Paolo Favaro



Timo Kehrer



Thomas Studer



Matthias Stürmer



Athina Tzovara



Kaspar Riesen

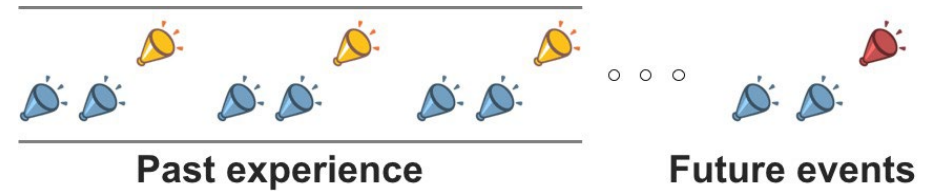


$u^b$

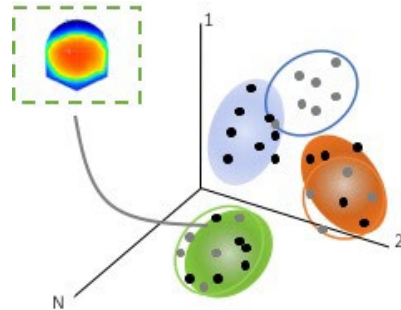
# Cognitive Computational Neuroscience (CCN)

Prof. Athina Tzovara

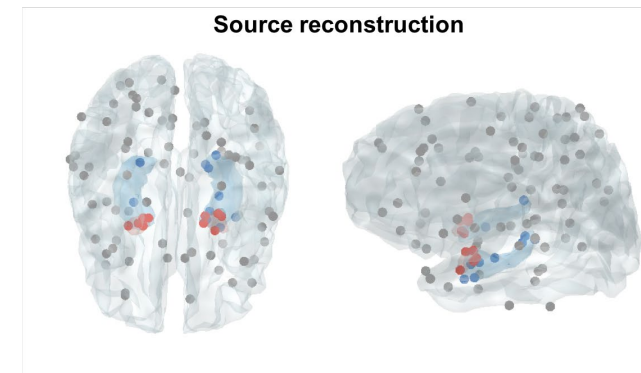
How do we detect patterns in our environment?



Machine learning techniques for analyzing neural data



Modeling neural activity

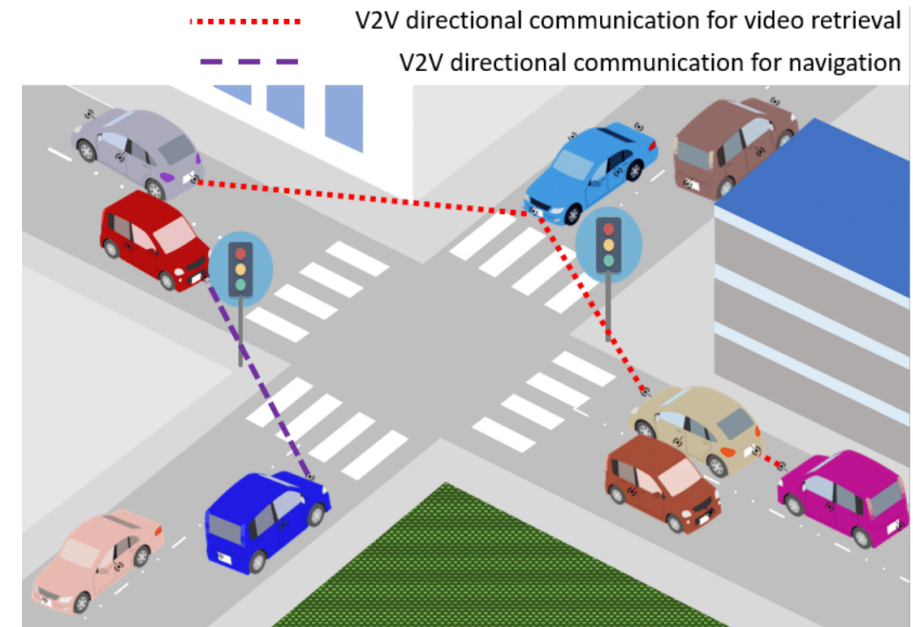
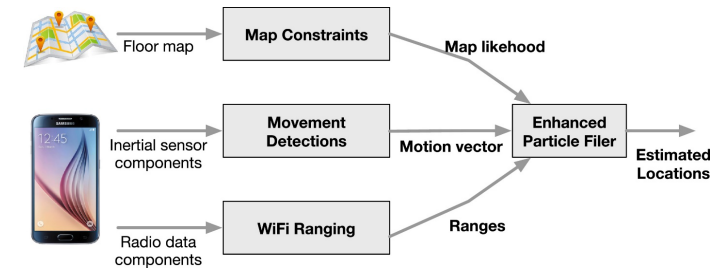


# Communication & Distributed Systems (CDS)

## Prof. Torsten Braun

### Research Topics

- Machine Learning in Networking, e.g. for mobility prediction
- Information- and Service-Centric Networking in highly mobile environments
- Indoor Localization



$u^b$

# Computer Graphics Group (CGG)

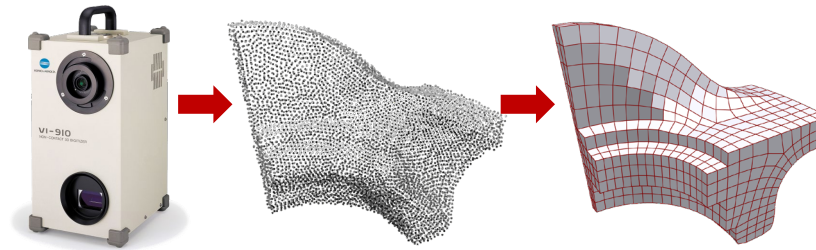
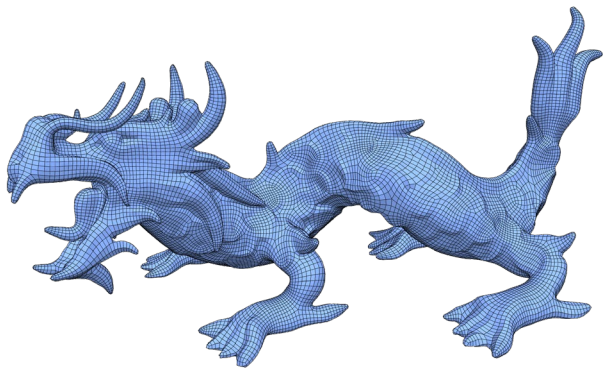
Prof. David Bommes

3D Reconstruction

3D Modeling

Fabrication

Applied Optimization



Geometry Processing

Mesh Generation



$u^b$

# Cryptology and Data Security (CRYPTO)

## Prof. Christian Cachin

- Cryptographic protocols
- Distributed algorithms

- Blockchains & distributed trust
- Consensus protocols
- Cloud-computing security

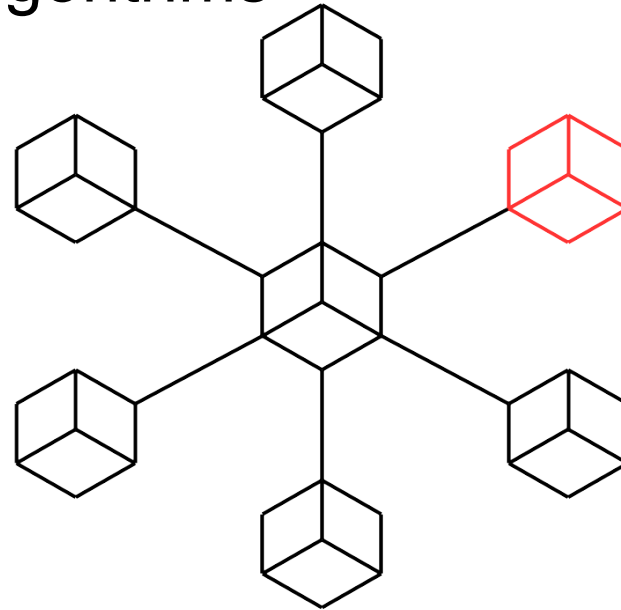
```

Algorithm 1 Conflict-free operation verification protocol (client  $C_i$ )
1: State
2:  $\delta \in \mathcal{O} \cup \{\perp\}$ : the operation being executed currently or  $\perp$  if no ope
3:  $c \in \mathbb{N}_0$ : sequence number of the last operation that has been confirm
4:  $H : \mathbb{N}_0 \rightarrow \{0,1\}^*$ : hash chain (see text), initially containing only  $H$ 
5:  $Z : \mathbb{N}_0 \rightarrow \mathcal{Z} \cup \{\perp\}$ : status map (see text), initially empty
6:  $s \in S$ : current state, after applying operations, initially  $s_0$ 

7: upon invocation  $o$  do
8:    $\delta \leftarrow o$ 
9:    $\tau \leftarrow \text{sign}_i(\text{INVOKE} \parallel o \parallel i)$  // last pending operatic
10:  send message  $[\text{INVOKE}, o, \tau]$  to  $S$ 

11: upon receiving message  $[\text{REPLY}, \text{Pend}]$  from  $S$  do // the
12:    $\text{Pend-other} \leftarrow ()$ 
13:    $\text{Pend-self} \leftarrow ()$  // list of
14:    $k \leftarrow 1$ 
15:   while  $k \leq \text{length}(\text{Pend})$  do
16:      $(o, j, \tau) \leftarrow \text{Pend}[k]$ 
17:      $l \leftarrow c + k$ 
18:     assert  $\text{verify}_j(\tau, \text{INVOKE} \parallel o \parallel j)$ 
19:     if  $H[l] = \perp$  then
20:        $H[l] \leftarrow \text{hash}(H[l-1] \parallel o \parallel j)$ 
21:     else assert  $H[l] = \text{hash}(H[l-1] \parallel o \parallel j)$ 
22:     if  $j = i \wedge k < \text{length}(\text{Pend}) \wedge Z[l] = \text{SUCCESS}$  then
23:        $\text{Pend-self} \leftarrow \text{Pend-self} \circ (o)$ 
24:     else if  $j \neq i$  then
25:        $\text{Pend-other} \leftarrow \text{Pend-other} \circ (o)$ 
26:        $k \leftarrow k + 1$ 
27:   // variables  $o, j$ , and  $l = c + \text{length}(\text{Pend})$  keep their values
28:   assert  $k \geq 1 \wedge o = \delta \wedge j = i$  // last pending operatic
29:   if not  $\text{conflicts}(s, \text{Pend-other}, \text{Pend-self} \circ (o))$  then
30:      $(s', \tau) \leftarrow F(s, \text{Pend-self} \circ (o))$  // compute resp
31:      $Z[l] \leftarrow \text{SUCCESS}$ 
32:   ..

```



$$g_i = \prod_{j=0}^{k-1} g_j^{\lambda_{j,i}^S}$$

$$\forall i, j \in [1, n], \forall Q_i \in \mathcal{Q}_i, \forall Q_j \in \mathcal{Q}_j, \forall F_{ij} \in \mathcal{F}_i^* \cap \mathcal{F}_j^* : Q_i \cap Q_j \not\subseteq F_{ij}.$$



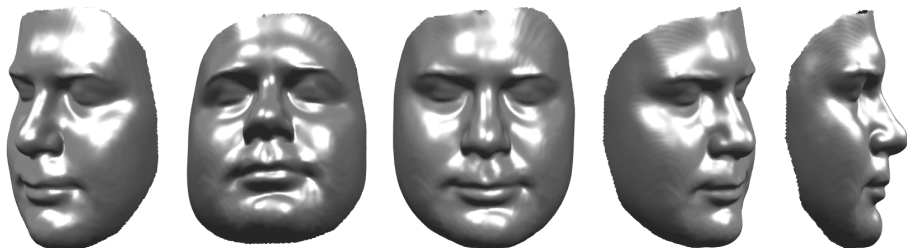
# $u^b$ Computer Vision (CVG)

Prof. Paolo Favaro

Computer vision



Computational photography

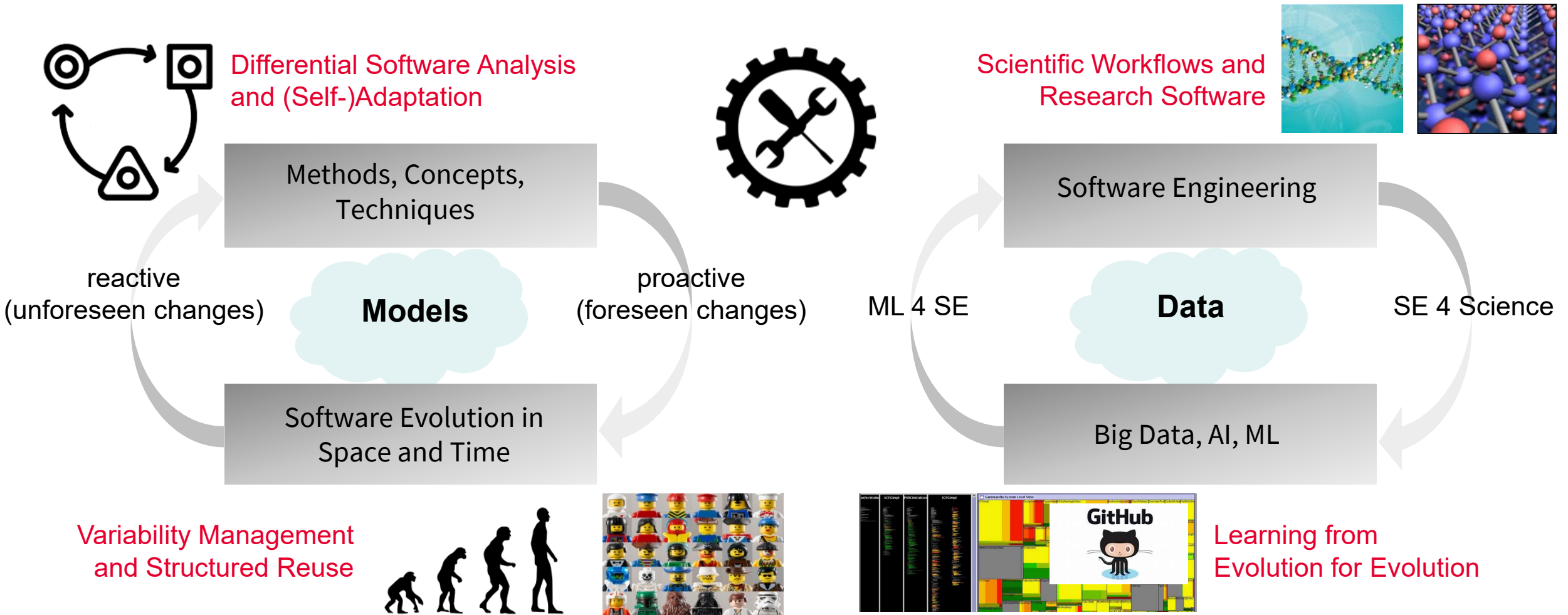


Artificial intelligence  
Machine learning

$u^b$

# Software Engineering (SEG)

Prof. Timo Kehrer



*u<sup>b</sup>*

# Research Center for Digital Sustainability

## PD Dr. Matthias Stürmer

Digital sustainability

Open-source software

Open data



Public procurement

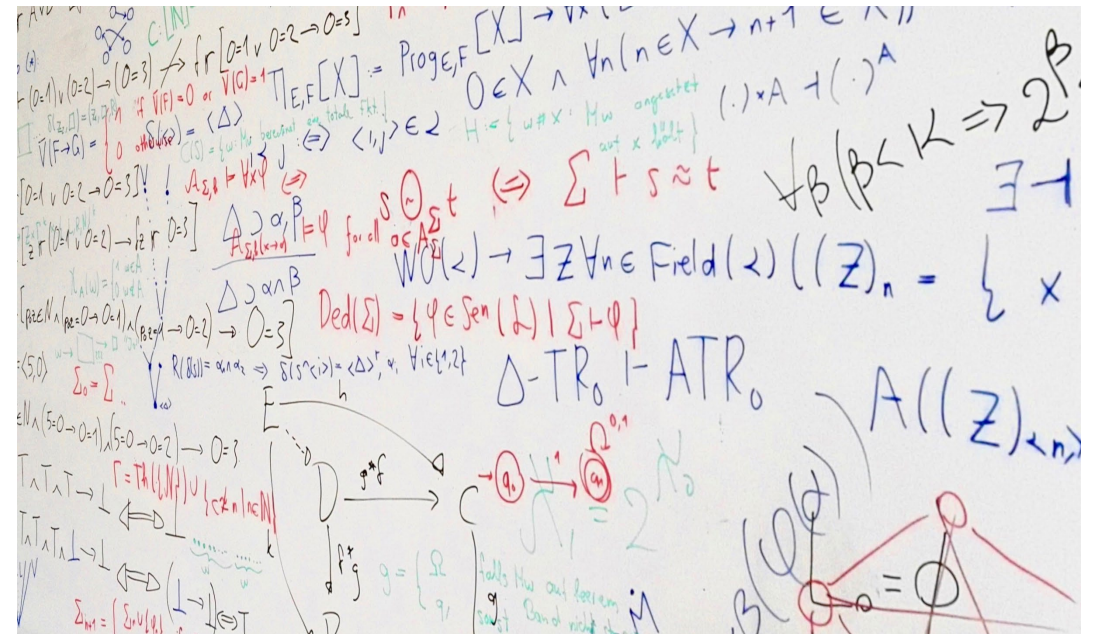
Open government

Smart cities

# Logic and Theory Group (LTG)

Prof. Thomas Studer

- Proof Theory
- Logics of Knowledge and Time
- Evidence-Based Reasoning
- Logic and Computation

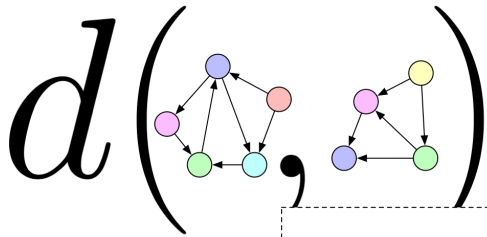




# Pattern Recognition Group (PRG)

PD Dr. Kaspar Riesen

Research fast and accurate algorithms for complex data structures  
 Compute, for instance, dissimilarities between graphs in polynomial time



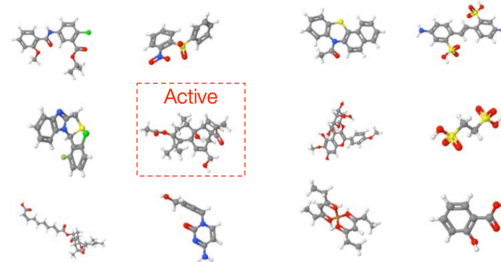
Model phenomena and entities with graphs to make crucial predictions

Predicting, for instance, temperature and discharge of water bodies in the future



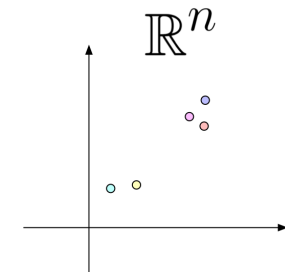
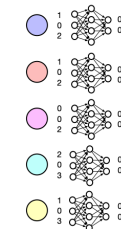
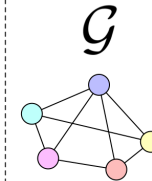
Find frequent and relevant patterns in large amounts of data

Determine, for instance, whether a certain molecule will show desirable activities (against a disease)



Research novel architectures for graph neural networks

Embed, for instance, graphs or networks in real vector spaces



*u*<sup>b</sup>

# Bachelorstudium Informatik

$u^b$

# European Credit Transfer System (ECTS)

- Jede Studienleistung wird in **ECTS Punkten** gemessen
- 1 ECTS Punkt: Für eine Veranstaltung sind während eines Semesters total 25-30 Arbeitsstunden aufzubringen (Präsenz, Übungen, Aufgaben, Prüfung)
- Die meisten Lehrveranstaltungen in Informatik entsprechen 5 ECTS. d.h., ca. 125-150 Arbeitsstunden
- Aufwand pro Woche und Lehrveranstaltung ca. 7.5-9.5 Stunden (125-150 Stunden, verteilt auf 14 Semesterwochen plus 1-2 Wochen Prüfungsvorbereitung und Prüfung)

# $u^b$ Anrechnung von ECTS

- ECTS werden nur angerechnet, sofern die Veranstaltung erfolgreich absolviert wurde: Die **Leistungskontrolle** wurde bestanden und benotet.
- **Vorlesungen**
  - Semesterschluss-Prüfungen, direkt nach dem Semester
  - Daten im Januar und im Juni
- **Seminare**
  - Obligatorisch ist ein Vortrag und/oder eine Arbeit



# $u^b$ Major-Studium Informatik

Sem	30 ECTS pro Semester	
1	Informatik 90 ECTS	1-4 Minor 90 ECTS (davon 30 in Mathematik)
2		
3		
4		
5		
6		

- 6 Semester zu je 30 ECTS, total 180 ECTS
- 90 ECTS im Major Informatik
- 90 ECTS in 1-4 Minors, davon **30 ECTS Mathematik obligatorisch** ( $\equiv$  GM30)

# $u^b$ Major-Studium Informatik

Sem	30 ECTS pro Semester	
1	Veranstaltungen in Informatik 70 ECTS	1-4 Minor 90 ECTS (davon 30 in Mathematik)
2		
3		
4		
5		
6	Bachelorarbeit + Seminar + Anl. wiss. Arbeiten, 20 ECTS	

- 70 ECTS aus Veranstaltungen in Informatik (frei wählbar)
- Jede Informatik-Veranstaltung gibt 5 ECTS, d.h., mindestens 14 Veranstaltungen

# $u^b$ Major-Studium Informatik

Sem	30 ECTS pro Semester	
1	Veranstaltungen in Informatik 70 ECTS	1-4 Minor 90 ECTS (davon 30 in Mathematik)
2		
3		
4		
5		
6	Bachelorarbeit + Seminar + Anl. wiss. Arbeiten, 20 ECTS	

- Abschluss durch das obligatorische Bachelorprojekt, welches umfasst:
  - Bachelorarbeit (10 ECTS)
  - Seminar (5 ECTS)
  - Anleitung zu wissenschaftlichen Arbeiten (5 ECTS)

# Veranstaltungen Informatik Bachelor

Semester 1 **Diskrete Mathematik** **Grundlagen der technischen Informatik** **Programmierung 1**

Semester 2 **Datenbanken** **Rechnerarchitektur** **Datenstrukturen und Algorithmen** **Programmierung 2**

Semester 3 **Computernetze** **Digitale Nachhaltigkeit** **Software Engineering**

Semester 4 **Berechenbarkeit und Komplexität** **Algorithmen, Wahrscheinlichkeit und Information**  
**Betriebssysteme** **Praktikum in Software Engineering**

Semester 5 **Maschinelles Lernen** **Computergrafik** **Mensch-Maschine-Schnittstelle**

Semester 6 **Bachelorarbeit** **Anleitung zu wissenschaftlichen Arbeiten** **Seminar**

Diese Abfolge ist empfohlen für das Bachelor-Studium in 6 Semestern.

Die Ordnung entspricht grob den Abhängigkeiten zwischen den Veranstaltungen.

# Veranstaltungen Informatik Bachelor

Semester 1	Diskrete Mathematik	Grundlagen der technischen Informatik	Programmierung 1	
Semester 2	Datenbanken	Rechnerarchitektur	Datenstrukturen und Algorithmen	Programmierung 2
Semester 3	Computernetze	Digitale Nachhaltigkeit	Software Engineering	
Semester 4	Berechenbarkeit und Komplexität	Algorithmen, Wahrscheinlichkeit und Information	Betriebssysteme	Praktikum in Software Engineering
Semester 5	Maschinelles Lernen	Computergrafik	Mensch-Maschine-Schnittstelle	
Semester 6	Bachelorarbeit	Anleitung zu wissenschaftlichen Arbeiten	Seminar	

Wie funktionieren Computersysteme?



# Veranstaltungen Informatik Bachelor

Semester 1	Diskrete Mathematik	Grundlagen der technischen Informatik	Programmierung 1	
Semester 2	Datenbanken	Rechnerarchitektur	Datenstrukturen und Algorithmen	Programmierung 2
Semester 3	Computernetze	Digitale Nachhaltigkeit	Software Engineering	
Semester 4	Berechenbarkeit und Komplexität	Algorithmen, Wahrscheinlichkeit und Information	Betriebssysteme	Praktikum in Software Engineering
Semester 5	Maschinelles Lernen	Computergrafik	Mensch-Maschine-Schnittstelle	
Semester 6	Bachelorarbeit	Anleitung zu wissenschaftlichen Arbeiten	Seminar	

Wie kann man mit Computern Probleme lösen?

# Veranstaltungen Informatik Bachelor

Semester 1	Diskrete Mathematik	Grundlagen der technischen Informatik	Programmierung 1	
Semester 2	Datenbanken	Rechnerarchitektur	Datenstrukturen und Algorithmen	Programmierung 2
Semester 3	Computernetze	Digitale Nachhaltigkeit	Software Engineering	
Semester 4	Berechenbarkeit und Komplexität	Algorithmen, Wahrscheinlichkeit und Information	Betriebssysteme	Praktikum in Software Engineering
Semester 5	Maschinelles Lernen	Computergrafik	Mensch-Maschine-Schnittstelle	
Semester 6	Bachelorarbeit	Anleitung zu wissenschaftlichen Arbeiten	Seminar	

Was können  
Computersysteme  
grundsätzlich und was nicht?

# Veranstaltungen Informatik Bachelor

Semester 1 **Diskrete Mathematik** **Grundlagen der technischen Informatik** **Programmierung 1**

Semester 2 **Datenbanken** **Rechnerarchitektur** **Datenstrukturen und Algorithmen** **Programmierung 2**

Semester 3 **Computernetze** **Digitale Nachhaltigkeit** **Software Engineering**

Semester 4 **Berechenbarkeit und Komplexität** **Algorithmen, Wahrscheinlichkeit und Information**  
**Betriebssysteme** **Praktikum in Software Engineering**

Semester 5 **Maschinelles Lernen** **Computergrafik** **Mensch-Maschine-Schnittstelle**

Semester 6 **Bachelorarbeit** **Anleitung zu wissenschaftlichen Arbeiten** **Seminar**

Wie interagieren Computer mit der menschlichen Umwelt?

# Veranstaltungen Informatik Bachelor

Semester 1	Diskrete Mathematik	Grundlagen der technischen Informatik	Programmierung 1	
Semester 2	Datenbanken	Rechnerarchitektur	Datenstrukturen und Algorithmen	Programmierung 2
Semester 3	Computernetze	Digitale Nachhaltigkeit	Software Engineering	
Semester 4	Berechenbarkeit und Komplexität	Algorithmen, Wahrscheinlichkeit und Information	Betriebssysteme	Praktikum in Software Engineering
Semester 5	Maschinelles Lernen	Computergrafik	Mensch-Maschine-Schnittstelle	
Semester 6	Bachelorarbeit	Anleitung zu wissenschaftlichen Arbeiten	Seminar	

Zum Abschluss: Einblick und  
Mitarbeit in aktueller  
Forschung

$u^b$

# Minor (Nebenfächer): Total 90 ECTS

- 30 ECTS Nebenfach Mathematik obligatorisch ("GM30")
- **Empfehlung:** Belegen Sie die Mathematik-Minor Kurse im 1. Drittel Ihres Studiums!
- **GM30** im Minor: Analysis 1 und Lineare Algebra 1 obligatorisch (je 8 ECTS), weitere frei, z.B. Analysis 2 und Lineare Algebra 2 (je 7)
- Mögliche Kombinationen
  - 90 ECTS Mathematik
  - 60 ECTS Mathematik + andere Minor (30 / 15+15)
  - 30 ECTS Mathematik + andere Minor (60 / 30+30 / 30+15+15)



*u<sup>b</sup>*

# Minor (Nebenfächer): Total 90 ECTS

- Typisch sind Minor-Fächer aus den Bereichen der
  - Philosophisch-naturwissenschaftliche Fakultät
  - Wirtschafts- und Sozialwissenschaftliche Fakultät
  - Philosophisch-historische Fakultät
  - Philosophisch-humanwissenschaftliche Fakultät
- Freie Leistungen (optional)
  - Aus dem gesamten Lehrangebot der Uni Bern, im Umfang von 15 ECTS

# Informatik als Minor-Studium

- Mögliche Varianten: Minor à 15, 30, 60 oder 90 ECTS
- Anzahl Veranstaltungen gemäss ECTS Punkten besuchen
- Veranstaltungen sind frei wählbar (aber: Logische Abhängigkeiten beachten!)
- Höchstens eine davon kann Master-Veranstaltung sein
  - Ausnahme: Minor zu 90 ECTS, dort auch mehr als eine
- Seminare nur nach vorherigem Antrag an die Studienleitung

$u^b$

# Anrechnung von Master-Vorlesungen

- Im Bachelor-Studium oder Informatik-Minor kann eine Veranstaltung aus der Informatik-Masterstufe besucht werden
  - Bei Minor zu 90 ECTS auch mehr als eine
- Regeln gemäss Joint Master in Computer Science der Universitäten BE, NE, FR
  - Separate Anmeldung nötig via Academia
- Weitere Informationen dazu
  - <https://mcs.unibnf.ch>

$u^b$

# Typische Organisation einer Vorlesung

- Keine spezielle Anmeldung erforderlich
- Unterrichtsmaterialien in Ilias: <https://ilias.unibe.ch>
- Akademische Viertelstunde
  - 16-18 Uhr bedeutet: Beginn um 16:15 Uhr
- Meist im ExWi Gebäude oder Areal Engehalde
- Üblich sind 2 Stunden Vorlesung + 1 Stunde Übungen
- Lösen der Übungen ist Voraussetzung für die Teilnahme an Prüfung
- Genaue Regelung je nach Vorlesung beachten!

*u*<sup>b</sup>

# Prüfungen: Anmeldung, Abmeldung

- Prüfungen am Semesterende, im Januar und im Juni
- Anmeldung via KSL (Kernsystem Lehre), <https://ksl.unibe.ch>
- Anmeldefristen laufen bis 2 Wochen vor der Prüfung (genaues Datum in KSL beachten!)
- Voraussetzung ist, dass eine Bedingung zu "Übungen" erfüllt ist



*u*<sup>b</sup>

# Prüfungen: Anmeldung, Abmeldung

- Allfällige Abmeldung via KSL (Termine beachten)
  - Wer ohne Abmeldung an der Prüfung nicht teilnimmt, gilt als durchgefallen (Note 1)
  - Wer wegen Krankheit nicht teilnehmen kann, muss die Studienleitung sofort informieren (und ein Arzt-Zeugnis zum frühest möglichen Zeitpunkt einreichen)

$u^b$

# Wiederholung von Prüfungen

- Wer eine Semesterschlussprüfung erstmals nicht besteht, hat Anrecht auf eine Prüfungswiederholung
- Zwei Möglichkeiten
  - Teilnahme Wiederholungsprüfung (meist mündlich); Information dazu über Dozent/in
  - Oder Vorlesung ein Jahr später nochmals besuchen (und die Übungen dazu lösen)
- Anmeldung zur Wiederholungsprüfung via KSL, innerhalb von drei Wochen nach Vorlesungsbeginn im nachfolgenden Semester

# $u^b$ Studienzeitverlängerung

- Die reguläre Studienzeit für das Bachelorstudium beträgt 6 Semester (3 Jahre)
- Wer 8 Semester (4 Jahre) überschreitet muss rechtzeitig (d.h. vor Ablauf dieser Frist) bei der Studienfachberatung Informatik (Frau Bettina Choffat) ein begründetes Gesuch um Verlängerung stellen!

# *u*<sup>b</sup> Weitere Informationen

- Studienfachberatung Informatik, Frau Bettina Choffat  
[studienberatung.inf@unibe.ch](mailto:studienberatung.inf@unibe.ch)
- Studiensekretariat des phil.-nat. Dekanats  
[https://www.philnat.unibe.ch/ueber\\_uns/dekanat/studiensekretariat](https://www.philnat.unibe.ch/ueber_uns/dekanat/studiensekretariat)
- Ihre Dozent/innen und Vorlesungsassistent/innen
- Mentors, [https://www.inf.unibe.ch/studium/mentors/index\\_ger.html](https://www.inf.unibe.ch/studium/mentors/index_ger.html)
- INF Website, <https://www.inf.unibe.ch/>
- Elektronisches Vorlesungsverzeichnis (KSL), <https://ksl.unibe.ch/>

$u^b$

# Auskünfte zu wichtigen Fragen

- Je wichtiger die Frage ist (Prüfung, Studiengestaltung), desto kompetenter sollte die Antwort sein
- Kompetente Quellen sind
  - Studienfachberatung Informatik
  - Studiensekretariat phil.-nat. Dekanat
  - Dozenten
- Wichtige Zusicherungen und Vereinbarungen unbedingt schriftlich bestätigen lassen
- Behauptungen wie "man hat mir gesagt ..." sind oft nicht gültig



# Wichtige Accounts

- Campus Account - <http://www.campusaccount.unibe.ch>
  - Gültig von Immatrikulation bis Austritt, erlaubt Zugriff auf Dienste der UniBE
  - Email: Meier, Alice (STUDENTS) <alice.meier@students.unibe.ch>
  - Login: Username: am19a193 Passwort: qrwHyvBCc7Ui
- SWITCH edu-ID - <http://edu-id.switch.ch>
  - Zugriff auf Dienstleistungen im Hochschulumfeld auch ausserhalb der UniBE
  - Bleibt über die Zugehörigkeit zur Universität hinaus gültig
  - [Verknüpfung mit Campus Account](#)
  - KSL, ILIAS, Unisport und vieles mehr ...

# Wichtige Dienste

- Achtung: Verschiedene Dienste verwenden unterschiedliche Benutzernamen

Dienstleistung	Studierende
<a href="#">Boris</a>	SWITCH edu-ID
<a href="#">eduroam</a>	campusaccount@unibe.ch
<a href="#">eForms</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">ILIAS</a>	SWITCH edu-ID
<a href="#">KSL</a>	SWITCH edu-ID
<a href="#">Mailclient</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">office365</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">Selfservice Studierende</a>	SWITCH edu-ID
<a href="#">SLS, Swiss Library Service Platform (swisscovery)</a>	SWITCH edu-ID
<a href="#">Serviceportal</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">Softwareshop</a>	SWITCH edu-ID
<a href="#">SUB Anmeldung</a>	SWITCH edu-ID
<a href="#">SwitchTube</a>	SWITCH edu-ID
<a href="#">Uniprint</a>	campusaccount
<a href="#">Unisport</a>	SWITCH edu-ID
<a href="#">VPN</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
Webmail	<a href="#">Webmail : outlook.office.com</a> E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">Webshop</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch
<a href="#">zoom</a>	E-Mail Adresse: vorname.nachname@students.unibe.ch

[https://www.unibe.ch/universitaet/campus\\_und\\_infrastruktur/rund\\_um\\_computer/campusbspaccount\\_zwnjswitch\\_edu\\_id/index\\_ger.html](https://www.unibe.ch/universitaet/campus_und_infrastruktur/rund_um_computer/campusbspaccount_zwnjswitch_edu_id/index_ger.html)

# Warum Informatik?

Handelszeitung Digital Switzerland  
 Unternehmen Management Invest Digital Switzerland Specials

Home > Digital Switzerland > Diesen 19 IT-Jobs gehört die Zukunft

WANDEL

## Diesen 19 IT-Jobs gehört die Zukunft

f t +



Arbeit im Serverraum: Mit dem Fortschritt ändern sich die Berufsbilder. Keystone

Die Wirtschaft verlagert sich zunehmend in den virtuellen Raum. Während alte Berufe verschwinden, wächst der Jobkatalog im IT-Bereich. 19 Tätigkeiten, die 2020 sehr gefragt sein werden.

Von Tomas Freres («Business Insider Deutschland») am 25.04.2016



«Früher wollte ich Architektin werden, heute setze ich meine Pläne im Web um.»

Isabel, Senior IT Business Analyst, UBS

[IT-dreamjobs.ch](http://IT-dreamjobs.ch)

Studiere Informatik – keine andere Branche bietet so grossartige Möglichkeiten!





Cloud computing

netzwoche

NEWS STORIES DOSSIERS VIDEO

NEWS

Lohnbuch des Kantons Zürich

## So viel verdienen Schweizer Informatiker 2019

Do 18.04.2019 - 08:57 Uhr  
 von [Oliver Schneider](#)




$u^b$

# Ihre Fragen



$u^b$

# Von Studierenden für Studierende

- Mentors, [https://www.inf.unibe.ch/studium/mentors/index\\_ger.html](https://www.inf.unibe.ch/studium/mentors/index_ger.html)
  - Marlene Kulowatz
  - Sascha Künzler
- Fachschaft Mathematik und Informatik der Uni Bern
  - <http://fsmib.ch>
  - <http://mib.unibe.ch>



Lehrveranstaltungen

Studienprogramme

FAQ

Studienfachberatung

Studieninteressierte

Informationstage

**Mentors**

Links

FAQ

Kontakt

Fachschaft

Evaluation

## Mentors

Als Mentors für das Informatikstudium möchten wir den StudienanfängerInnen die Möglichkeit eines persönlichen Kontakts zu höheren Semester anbieten. Wir studieren alle Informatik an der Universität Bern.

In erster Linie wollen wir bei den vielen kleinen und grösseren Fragen als erste Ansprechpartnerinnen zur Verfügung stehen. Dieser Unterbereich der Website wird von uns gepflegt und nicht vom Institut für Informatik. Hier findet ihr Neuigkeiten und Tipps rund ums Studium, sowie Kontakte und Antworten auf die häufigsten Fragen.

Falls ihr Fragen zum Studium habt, zögert nicht uns eine E-Mail zu schreiben. Auch Wünsche und Vorschläge sind natürlich stets willkommen.

*Marlene Kulowatz und Sascha Künzler (HS 2022)*



# Fachschaft Mathematik und Informatik

vorstand@fsmib.ch fsmib.ch

## Programm HS 2023



**September**

DO 21.09. Billard-Abend  
MI 27.09. Aktion für Podcasts

**Oktober**

DI 03.10. Tichu-Karaoke-Stammtisch  
MI 04.10. GV  
DO 19.10. Brätle mit bsi und fpa  
MI 25.10. Zühlke Workshop

**November**

MI 01.11. Halfwaythrough  
DI 07.11. Tichu-Karaoke-Stammtisch  
DO 16.11. Insight-Event  
DO 23.11. Movienight

**Dezember**

DI 05.12. Tichu-Karaoke-Stammtisch  
MO 11.12. M&S Glühwein  
DO 14.12. Adventsessen mit ipt und fpa

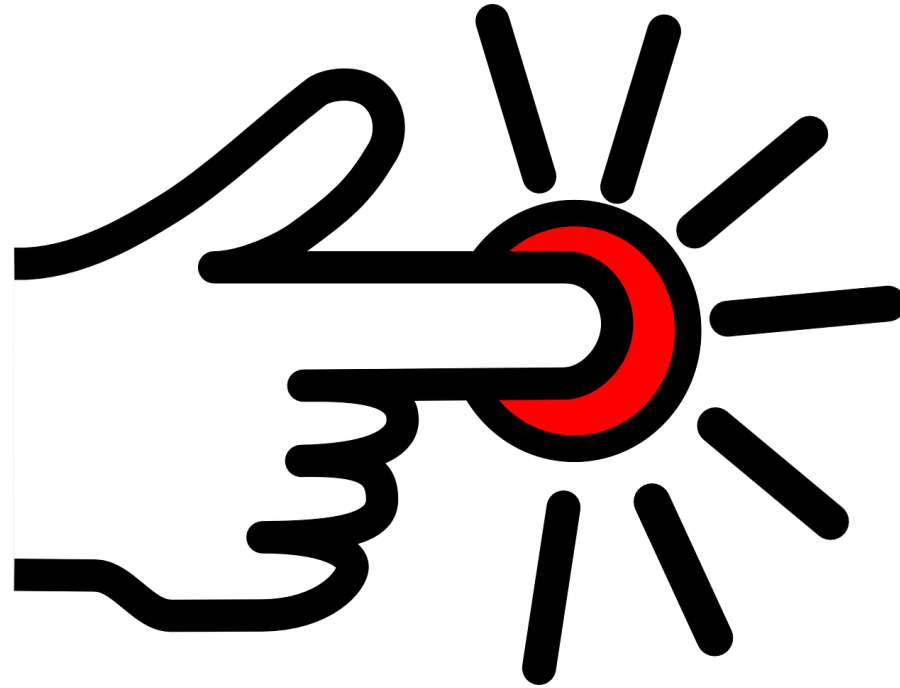
# FACHSCHAFT MATHEMATIK UND INFORMATIK



**MIB**  
Mathematik und Informatik

*u<sup>b</sup>*

# Viel Erfolg im Bachelorstudium!



[www.inf.unibe.ch](http://www.inf.unibe.ch)