ABOUT THE TEACHER

MAARTEN VAN STEEN

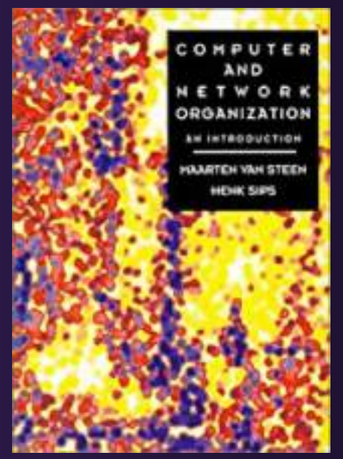
UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

INTRODUCTION
MAARTEN VAN STEEN

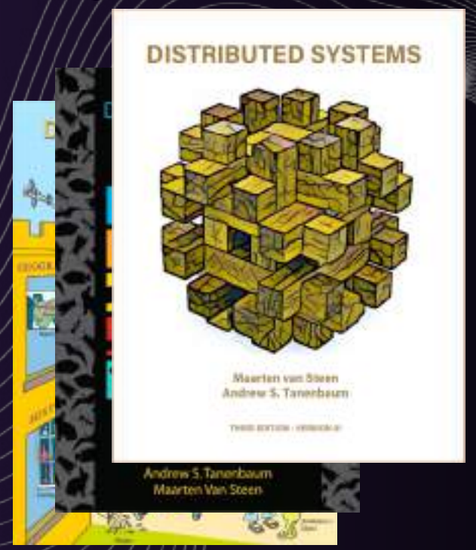
- Professor in Computer Science
- Scientific Director of Digital Society Institute

- Large-scale distributed computer systems
- Wireless sensor networks for crowd monitoring

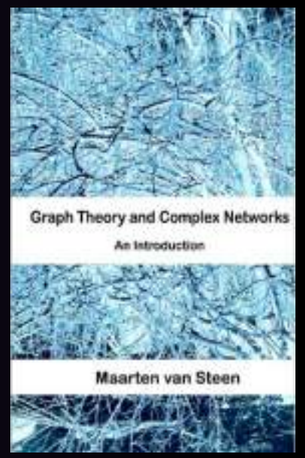
Computers & networks



Large-scale distributed systems



Complex networks



UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

INTRODUCTION

- Professor in Computer Science
- Scientific Director of Digital Society Institute

- Large-scale distributed computer systems
- Wireless sensor networks for crowd monitoring

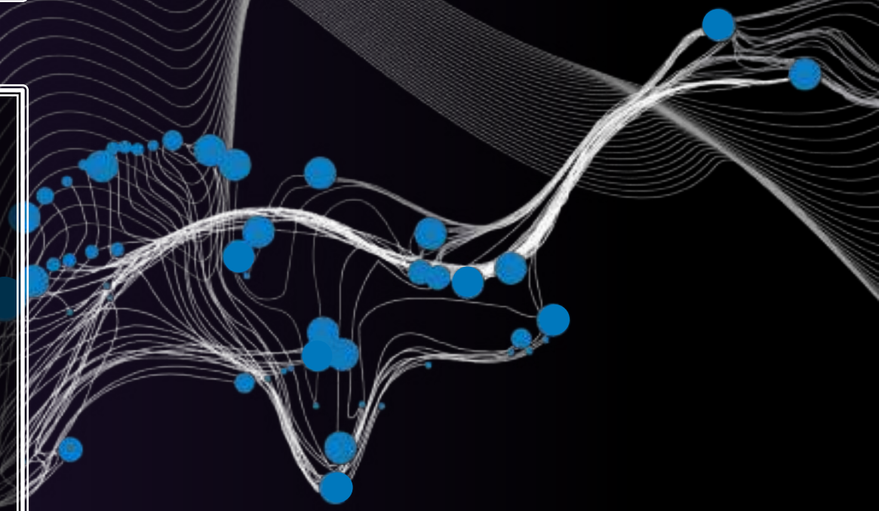
UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

DISTRIBUTED SYSTEMS
A LESS TRADITIONAL COURSE

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

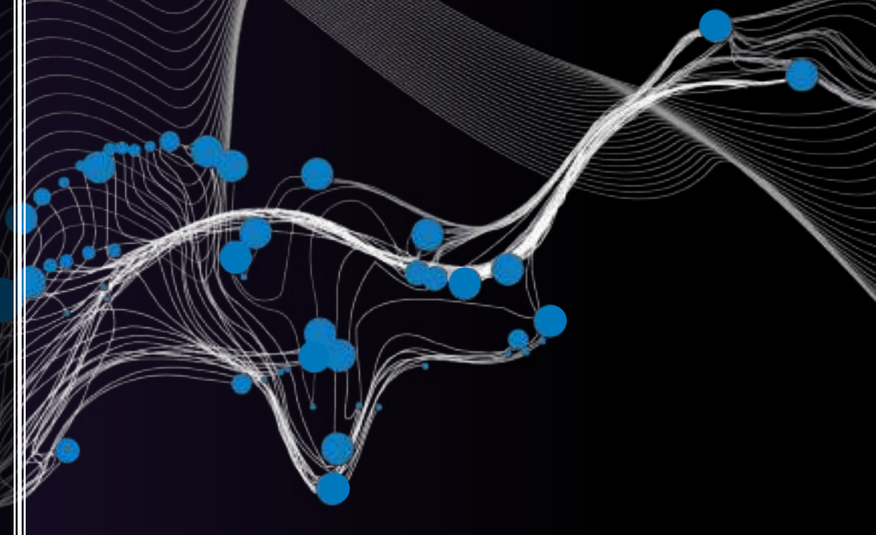
- **WHAT'S THE ISSUE?**
  - On effective knowledge transfer
  - On the complexity of distributed systems
  - On training cognitive competences

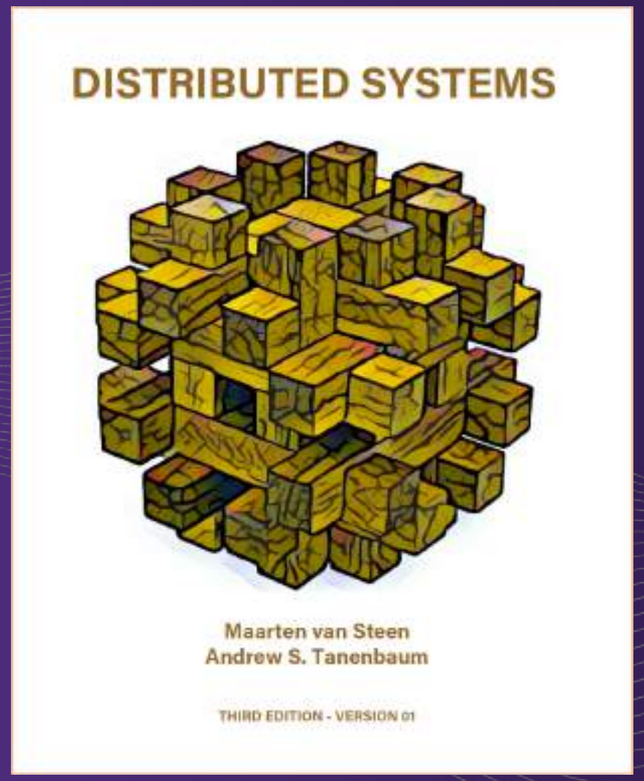
- **WHAT'S THE SOLUTION?**
  - You will be trained in understanding complex problems in distributed systems
  - Training by explaining: you cannot explain a problem or solution you do not understand well
  - If you understand one complex problem of today, you'll be able to understand problems of tomorrow



UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

- **APPROACH**
  - I will discuss briefly a number of controversial problems in distributed systems
  - In small groups, you'll dig into a problem of your choice
  - You will train yourself in critical thinking...
    - ... by explaining solutions to each other
    - ... by explaining solutions to the entire group
  - You will be supervised in your explanation sessions (group-wise appointments)

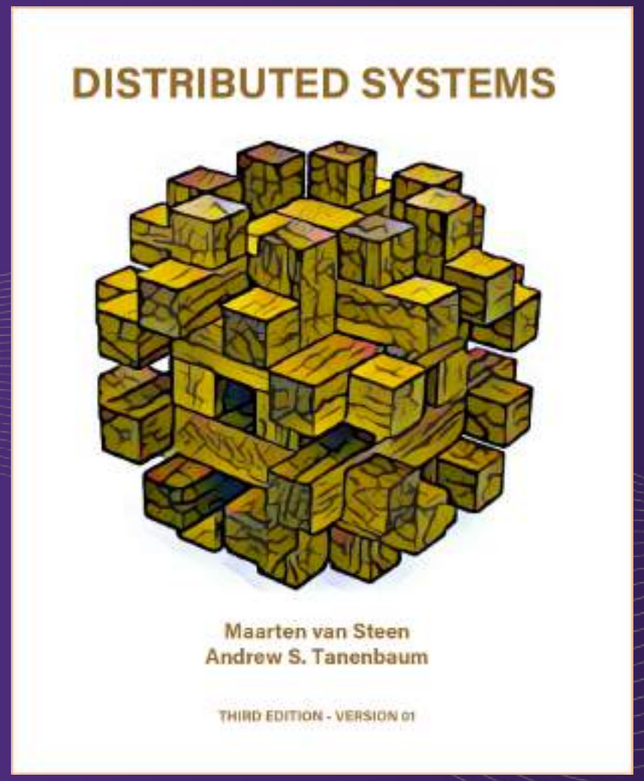


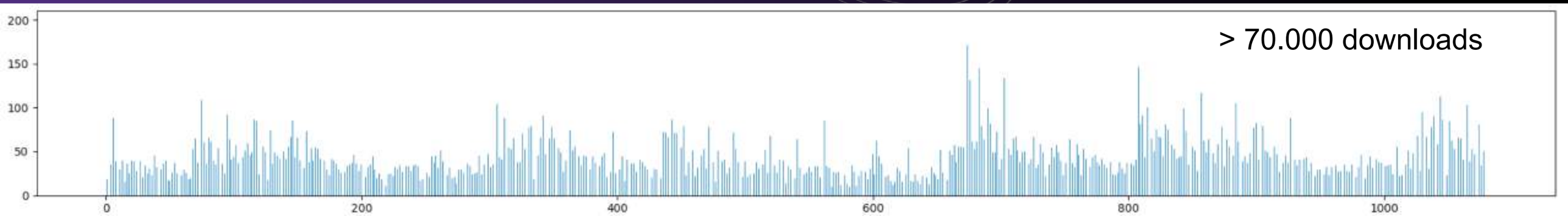


www.distributed-systems.net

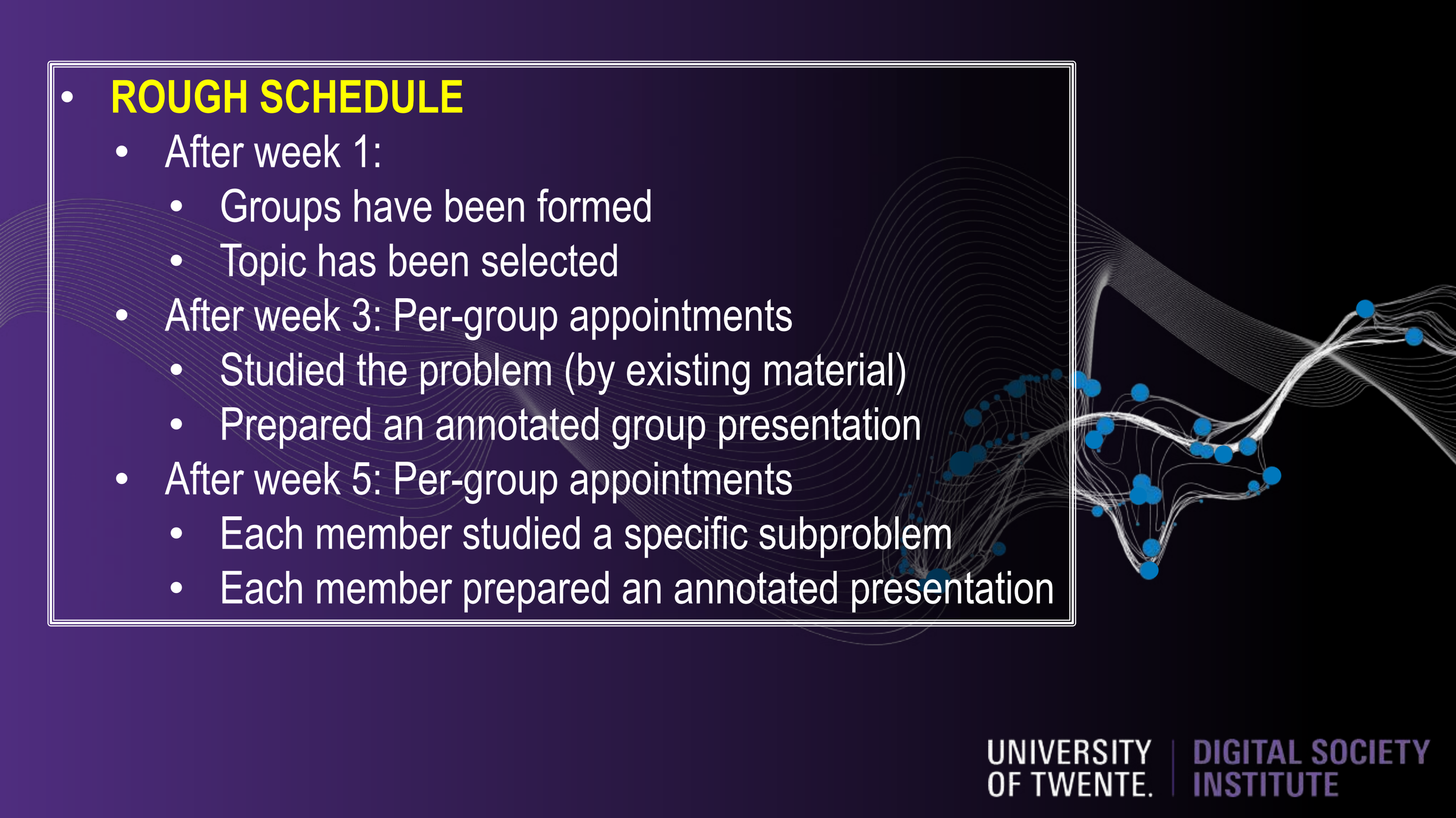
canvas

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

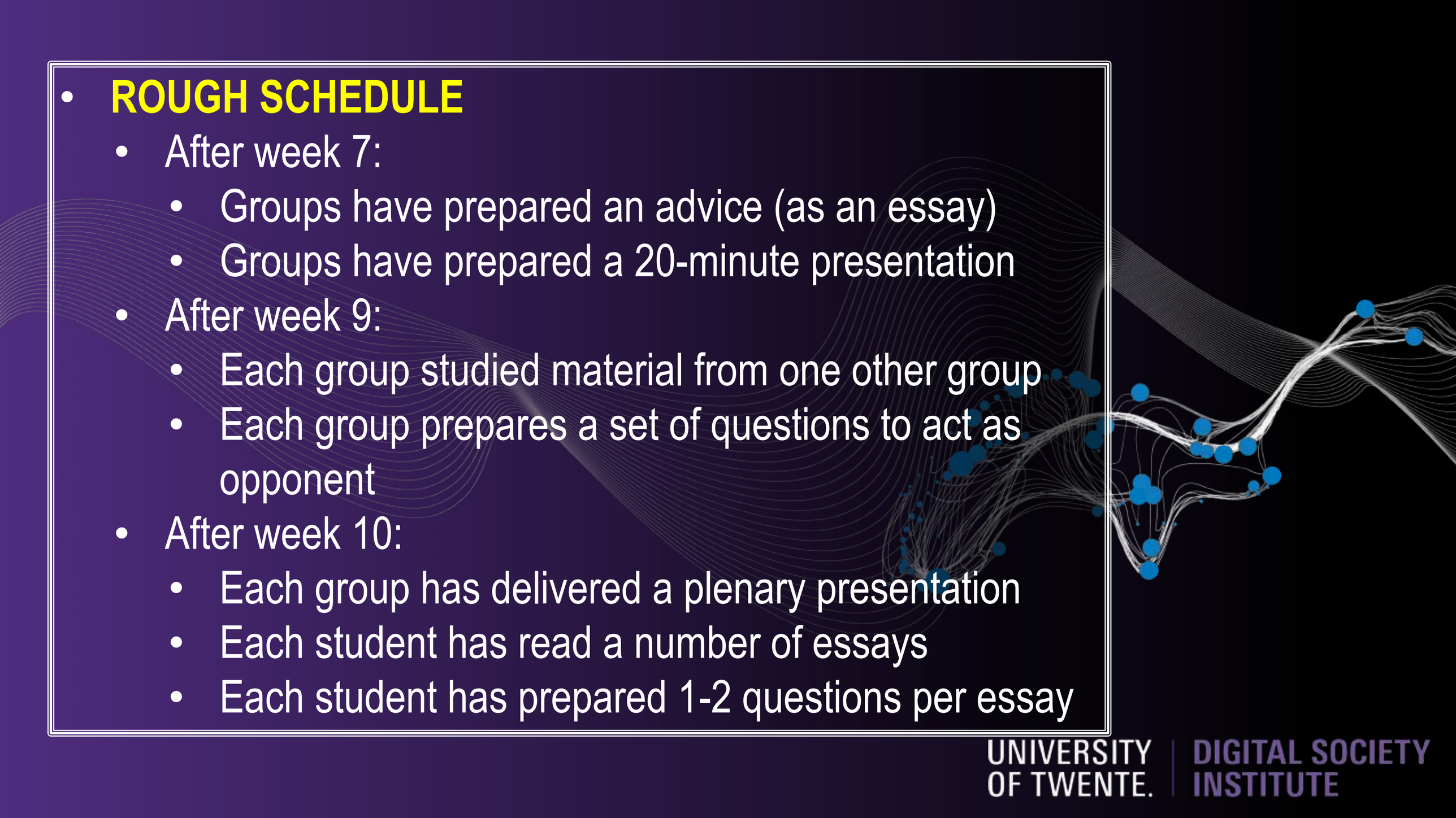

DISTRIBUTED SYSTEMS
Maarten van Steen
Andrew S. Tanenbaum
THIRD EDITION - VERSION 01


200
150
100
50
0
0
200
400
600
800
1000
> 70.000 downloads

- **ROUGH SCHEDULE**
  - After week 1:
    - Groups have been formed
    - Topic has been selected
  - After week 3: Per-group appointments
    - Studied the problem (by existing material)
    - Prepared an annotated group presentation
  - After week 5: Per-group appointments
    - Each member studied a specific subproblem
    - Each member prepared an annotated presentation



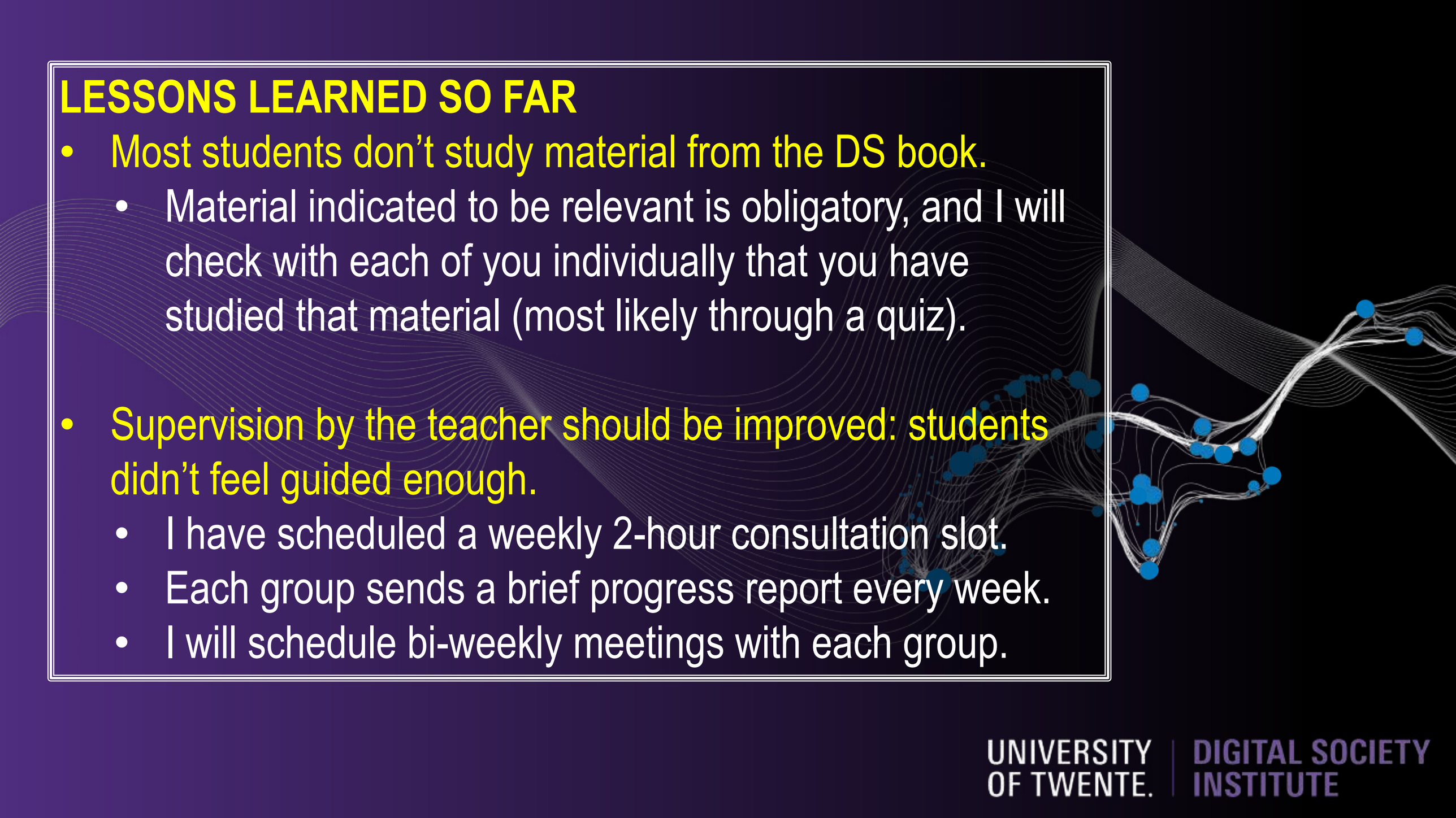
- **ROUGH SCHEDULE**
  - After week 7:
    - Groups have prepared an advice (as an essay)
    - Groups have prepared a 20-minute presentation
  - After week 9:
    - Each group studied material from one other group
    - Each group prepares a set of questions to act as opponent
  - After week 10:
    - Each group has delivered a plenary presentation
    - Each student has read a number of essays
    - Each student has prepared 1-2 questions per essay



LESSONS LEARNED SO FAR

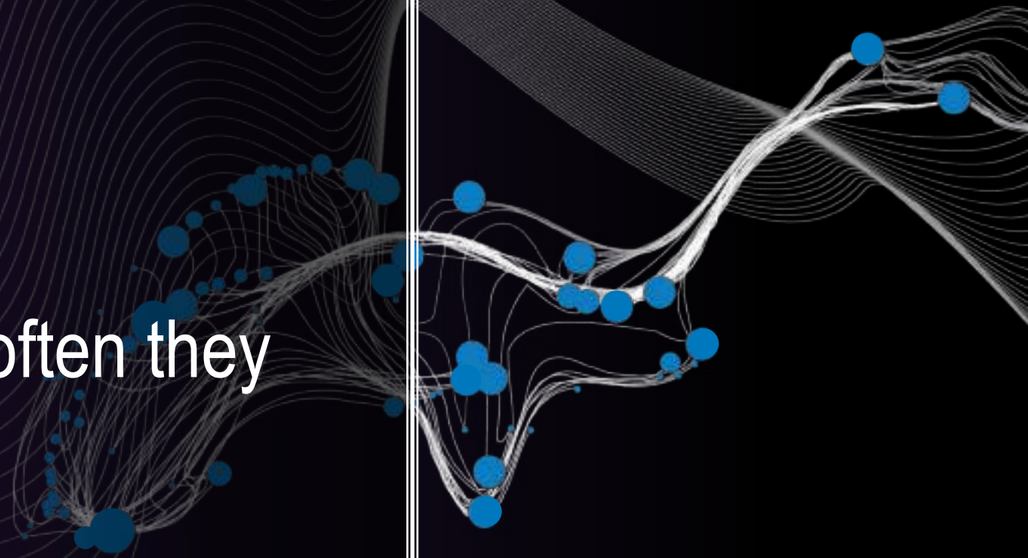
- Most students don’t study material from the DS book.
  - Material indicated to be relevant is obligatory, and I will check with each of you individually that you have studied that material (most likely through a quiz).

- Supervision by the teacher should be improved: students didn’t feel guided enough.
  - I have scheduled a weekly 2-hour consultation slot.
  - Each group sends a brief progress report every week.
  - I will schedule bi-weekly meetings with each group.



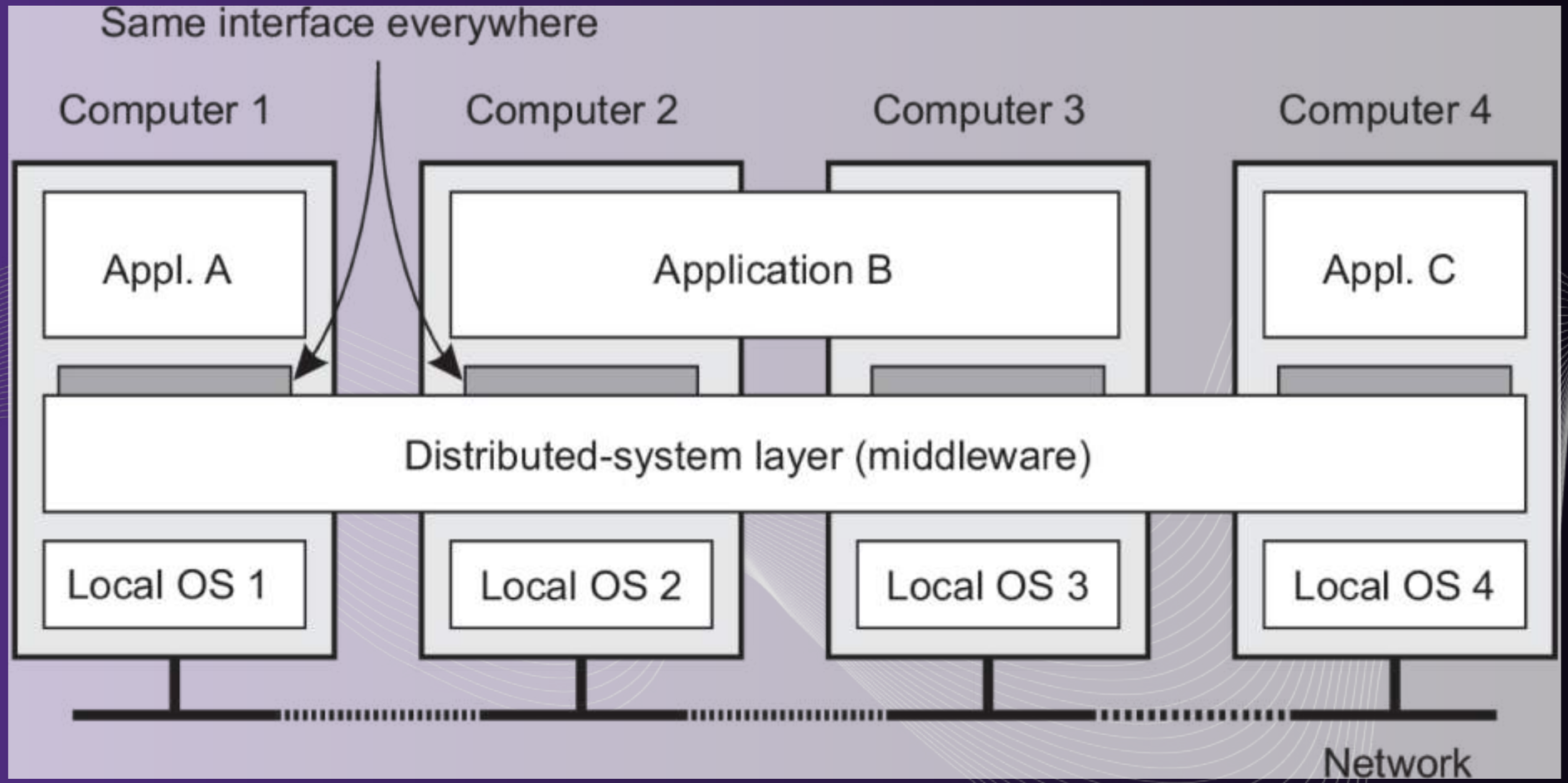
A FEW TIPS

- There are guidelines for essays on the site.
- There are three journals that may help find your way:
  - ACM Computing Surveys
  - IEEE Communications Surveys & Tutorials
  - Computer Science Review
- Google scholar
  - Check articles, their publication date, how often they have been cited, and by whom.
- Keep a good pace, don’t postpone things.
  - If you get stuck, you get in touch!



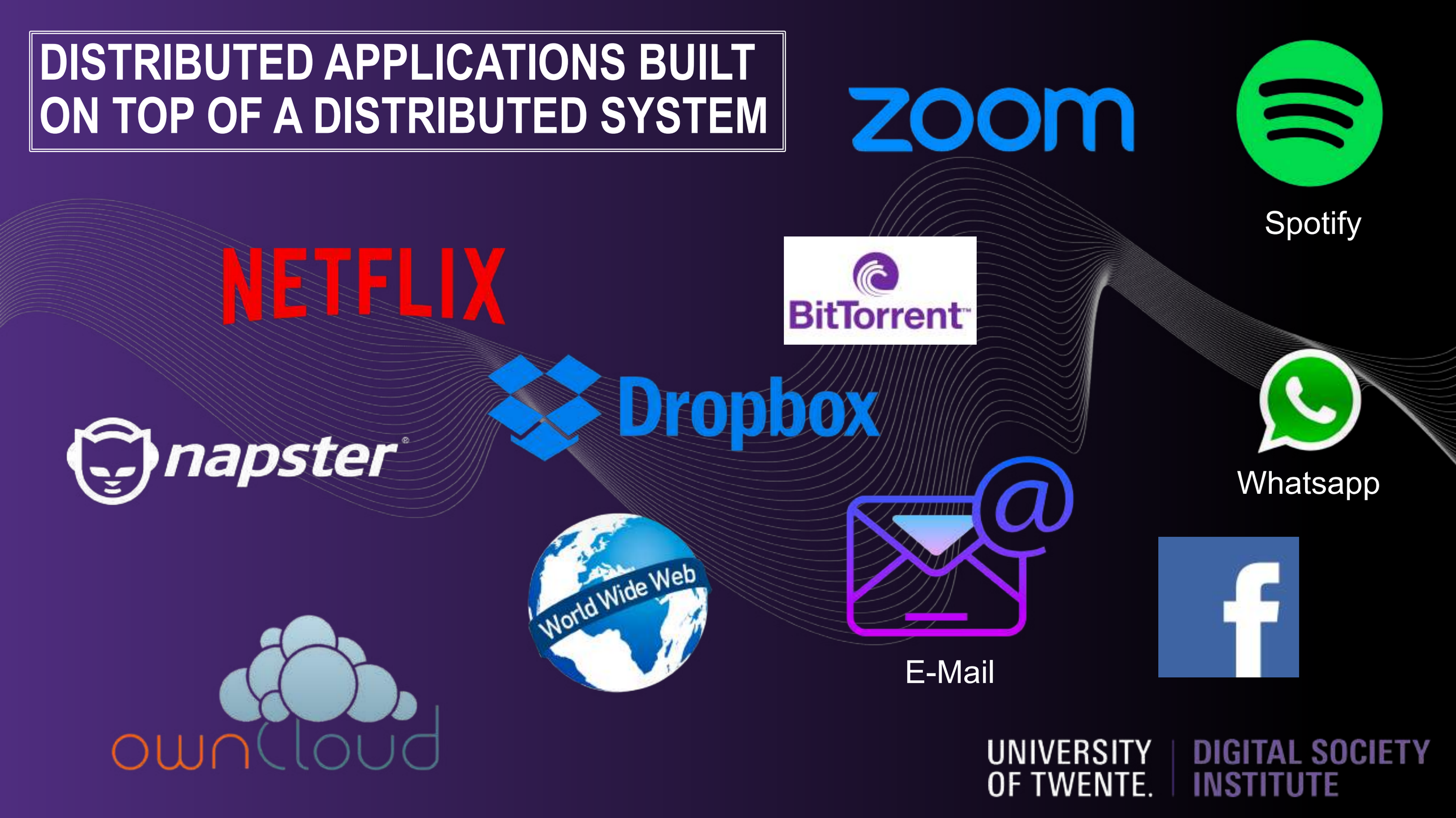
DISTRIBUTED SYSTEMS INTRODUCTION

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE



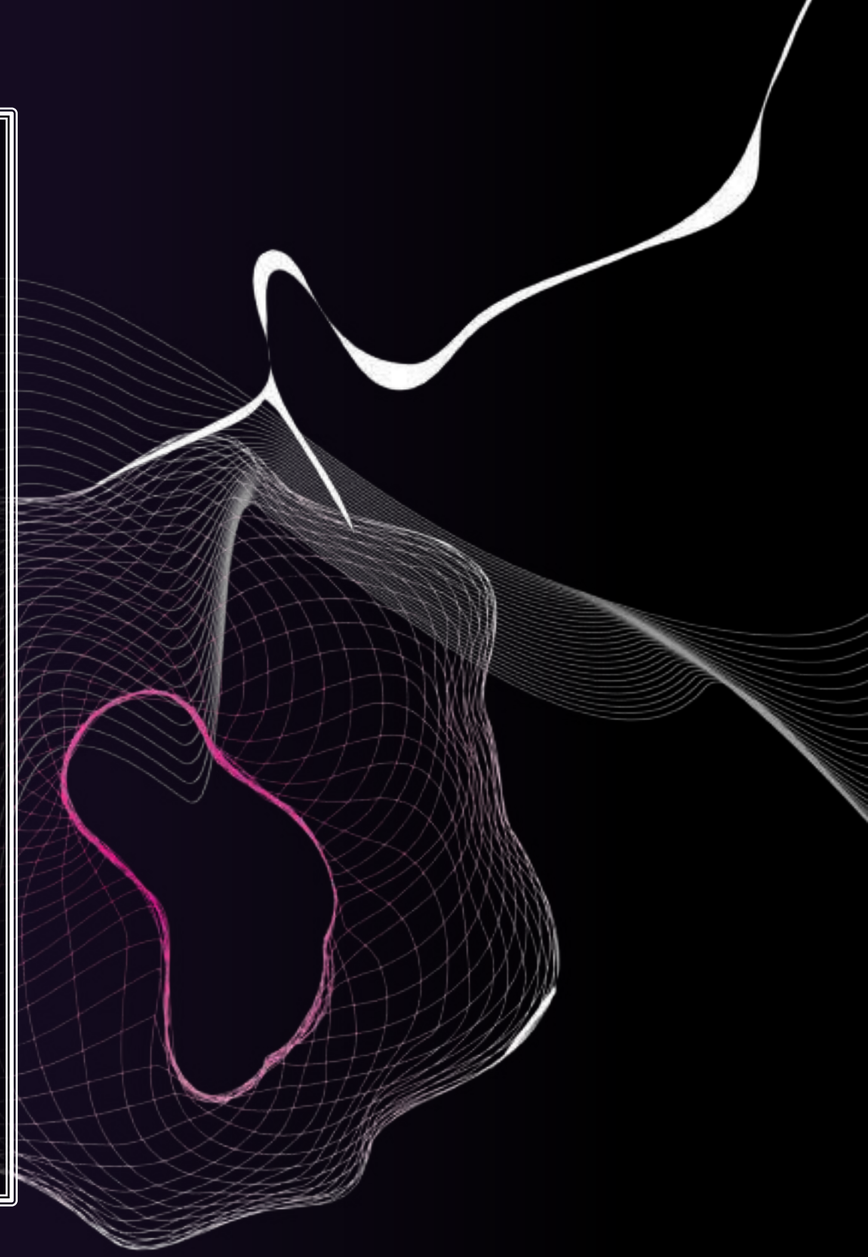
The glue between applications and operating systems

DISTRIBUTED APPLICATIONS BUILT ON TOP OF A DISTRIBUTED SYSTEM

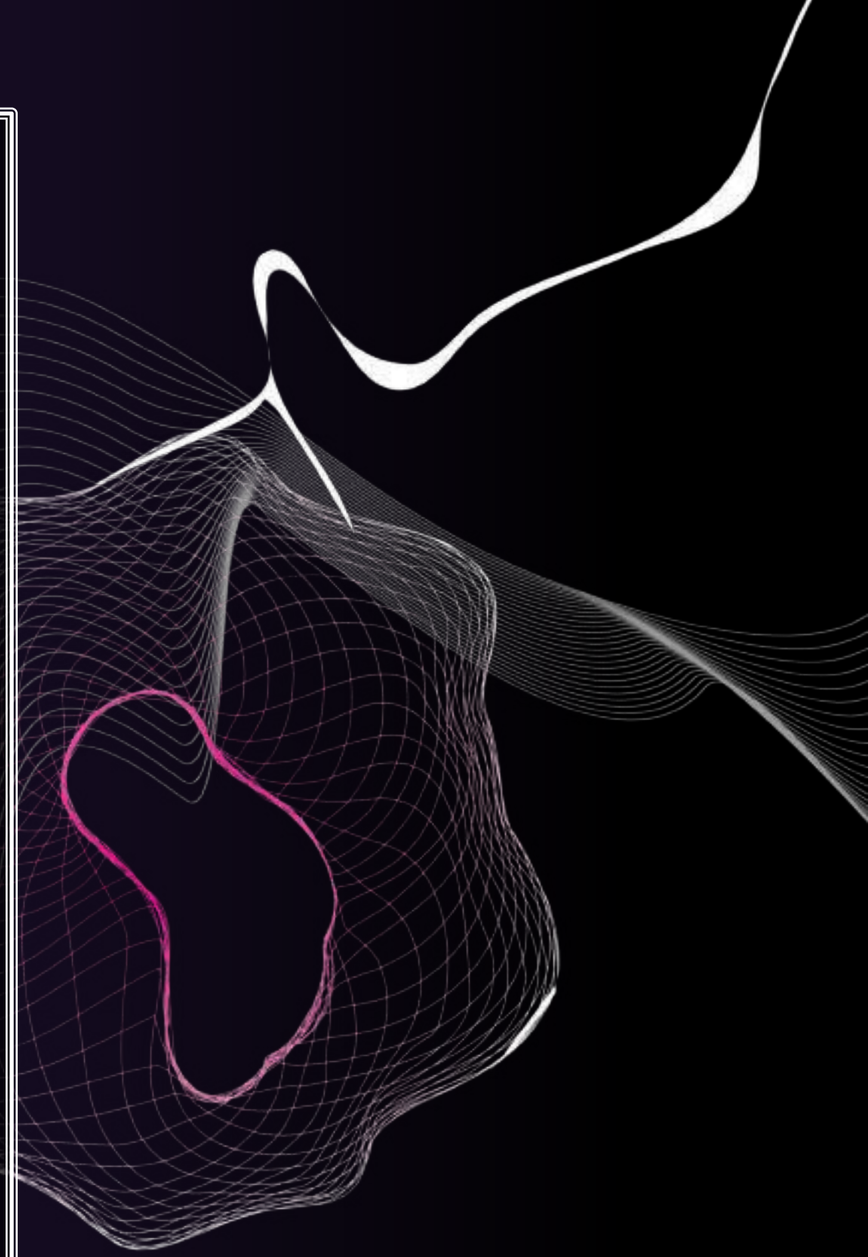


UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

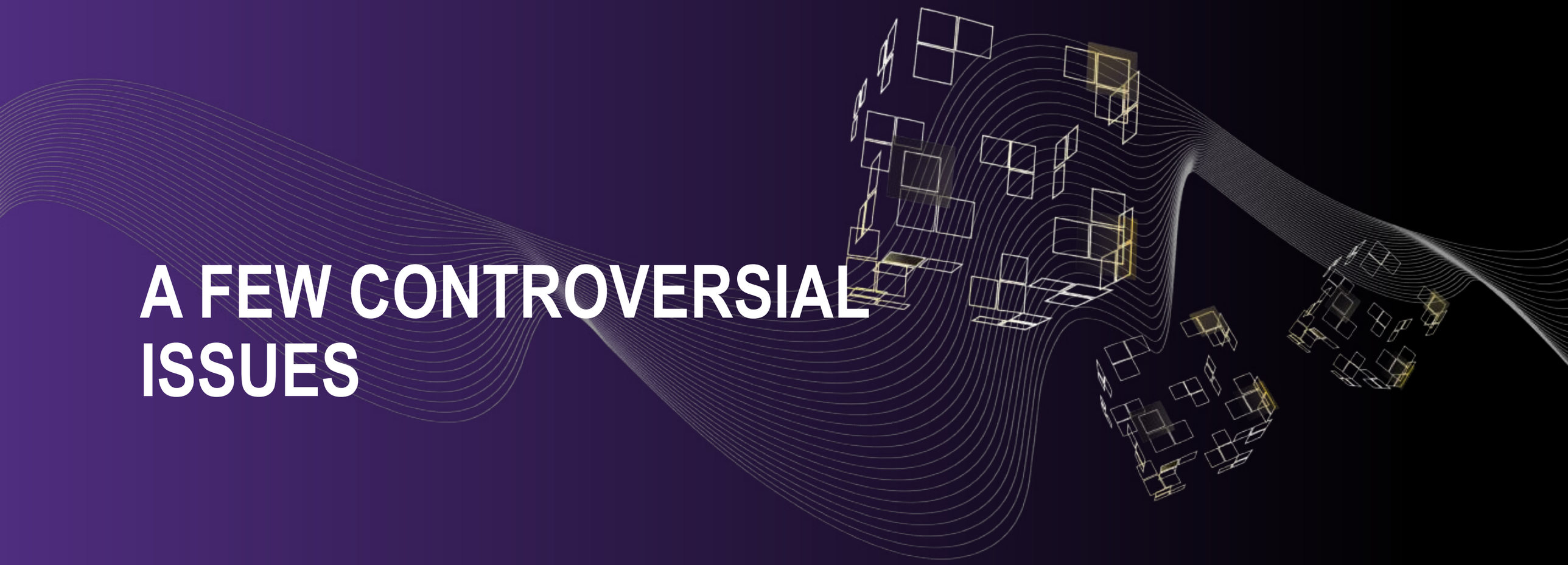
- **IN THE MIDDLE LAYER**
  - Communication services: calling procedures at remote locations (RPC), multicasting, content-based matching of senders and receivers.
  - Reliability services: ensuring consistent backups or hot stand-by's.
  - Resource sharing: enabling easy-to-deploy cloud-based file sharing (e.g., Dropbox) or peer-to-peer networking.
  - Security services: single sign-on, mixing unix & windows file systems (Samba).



- **NO FREE LUNCH**
  - Scalability: in size, across locations, across administrative boundaries
  - Data consistency in the face of replication: keeping copies the same
  - Fault tolerance: a distributed system is subject to partial failures, no all-or-nothing failures
  - Distribution transparency: what you need to know about the distribution of the system
  - Security: distribution of processes and data, notably across multiple organizations, increases vulnerability

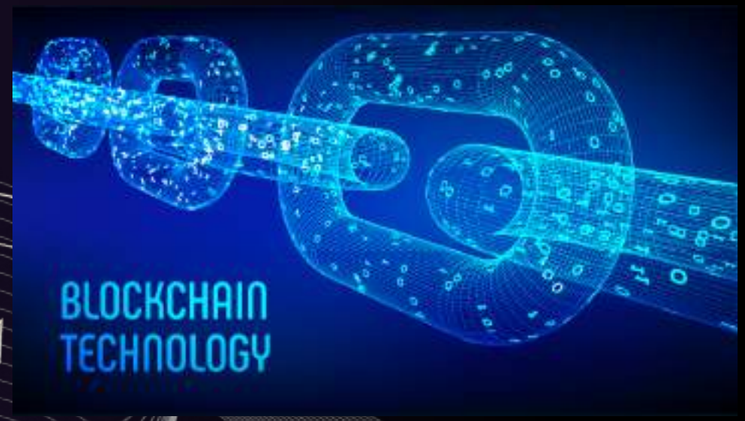


A FEW CONTROVERSIAL ISSUES



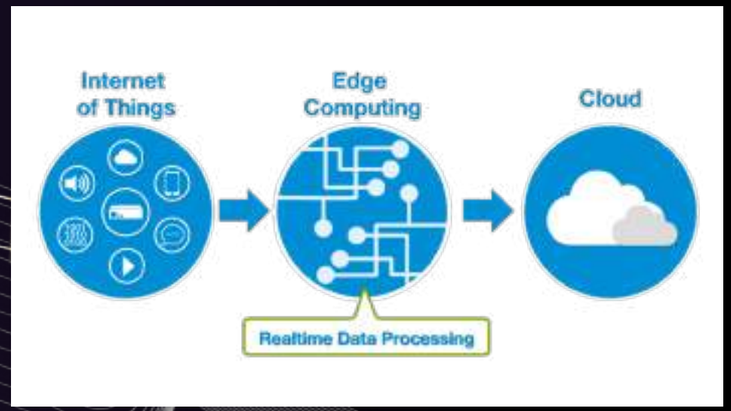
UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

- **BLOCKCHAINS**
  - A lot of buzz: it allows transaction processing without the use of a trusted third party (?)
    - fully decentralized: no TTP
    - everyone agrees on a transaction
    - scalable
  - There are some issues:
    - complete agreement, scalability, processing capability (speed), total democracy may not be possible to combine.



- **OPEN PEER-TO-PEER SYSTEMS**
  - A lot of buzz: publicly sharing files across the Internet without a centralized service: disruption by tearing down the central service is not going to help you (e.g., Pirate Bay).
  - There are some issues:
    - Sybil attack: creating so many malicious peers such that they control the network.
    - Eclipse attack: creating enough malicious peers to isolate a single benign peer.
    - Effect: the network is significantly disrupted.

- **EDGE COMPUTING**
  - A lot of buzz: with the Internet-of-Things exponentially growing, we deploy an infrastructure at the edge of the Internet. This opens a whole (research) can of worms: how to balance what's going on at the edge and what in the cloud?
  - There are some issues:
    - There are few compelling reasons to run things at the edge.
    - Perhaps the problem is much smaller than what we (are suppose to) envisage.



- **VIRTUAL MACHINES VS CONTAINERS**
  - A lot of buzz: when Docker entered the game of cloud computing, their containers turned out to be the lightweight (and thus much better) competitors of virtual machines.
  - There are some issues:
    - Lots of people claim that containers are much more efficient than virtual machines, but is this really true?
    - The next claim is that they are also highly portable. When you understand some technicalities, you may doubt this claim.

- **THE BEAUTY OF PUBLISH-SUBSCRIBE**
  - A lot of buzz: instead of direct communication, we can match a sender and a receiver based on the content of messages: decoupling to the max.
  - There are some issues:
    - Scalability can be easily at stake: how to do decentralized and scalable matching if you don't know where to match?
    - Secure communication between decoupled and mutually independent processes?


MAYBE I COULD PERMANENTLY WORK AT HOME.
NO PROBLEM. I JUST NEED A FEW THINGS FROM YOU TO MAKE SURE YOU ARE WORKING.
SUCH AS?
WELL, OBVIOUSLY, I NEED FREQUENT STATUS REPORTS.
SOUNDS REASONABLE.
AND I'LL NEED TO TRACK YOUR KEYSTROKES AND YOUR PHONE'S LOCATION.
WOW. WELL, OKAY. I GUESS I CAN GET USED TO THAT IN RETURN FOR MY FREEDOM TO WORK AT HOME.
NOW THAT I'VE LOOSENED YOU UP, LET'S TALK ABOUT FITTING YOU FOR A BODY CAM.
Twitter: @scottadamssays
© 2020 Scott Adams, Inc./Dist. by Andrews McMeel
Dilbert.com
7-5-20