Glossary of info1111

Week 1

1. <don’t stop learning!!!>

   •We will teach you core skills, but…

   •Important for your future careers, that you can grow and develop...

   •This is more important in IT than almost any other area!
2. <what info1111 learning>Topics covered

   •Professionalism

   •Teamwork

   •Tools

   •Pathways (Majors and career options)
3. <assessment-levels>

   Level1(Satisfactory):Demonstrated the core required content and skills to the level required to move on to subsequent units

   Level2(Advanced):(1)Succeeded at Level 1 requirements(2)Demonstrated similar knowledge and skills to Level 1, but to a significant degree of mastery

   level3(Expert):(1)Succeeded at Level 2 requirements(2)Demonstrated knowledge and skills beyond the core requirements

   level4(Outstanding):(1)Succeeded at Level 3 requirements(2)Demonstrated exceptional knowledge and skills beyond the core requirements
4. <Uos overview>

   -Source: Different types of sources of help are accepted for different academic levels (Junior, Intermediate, Senior, Postgraduate)

   -Nature of Help: Different types of help are accepted for different types of assessment----------Two slides explaining “Source” and “Nature” of help allowed

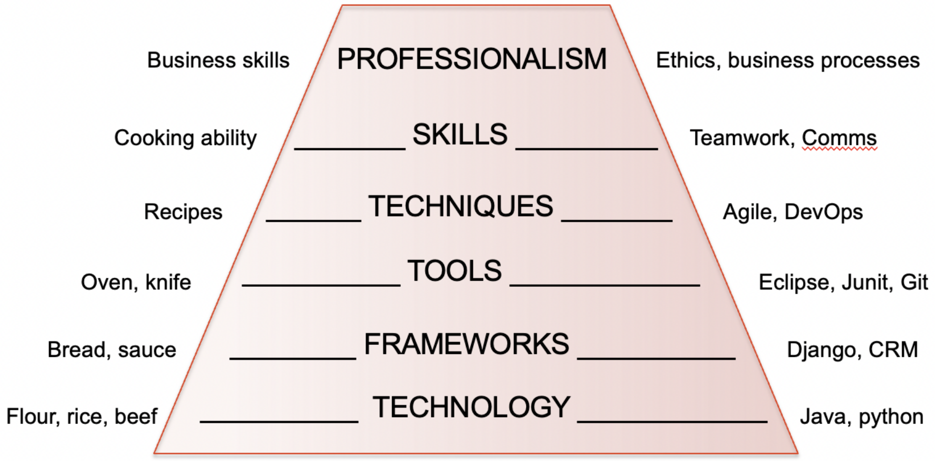
   –You can adjust the chart to suite the academic integrity requirement for your assessments.

   5.<Recap>•IT Professions – varied, require wide range of skills to succeed.

   •Future as an IT professional – rapidly changing environment requires continuous learning.

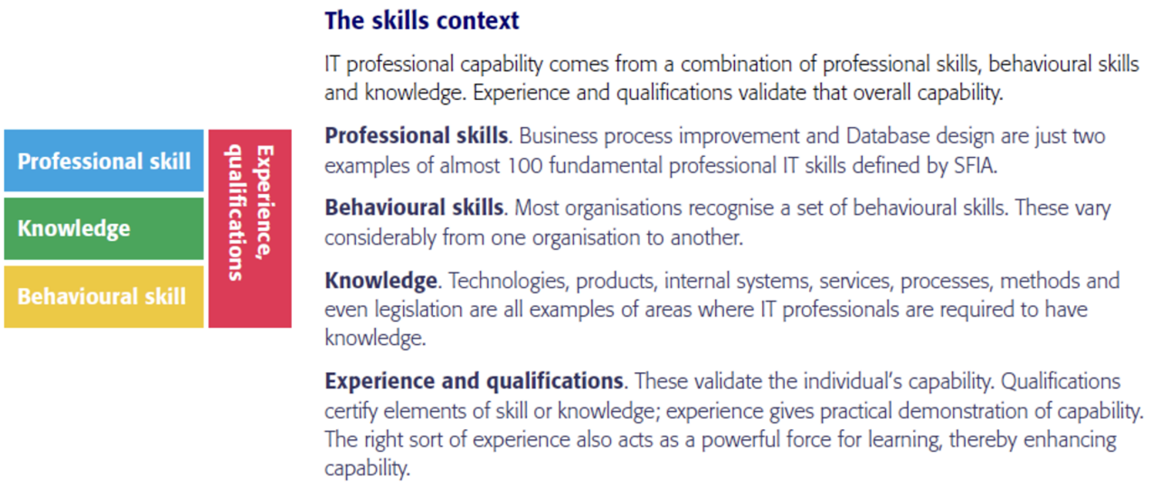
Week 2

1.<Professionalism>Technical capability is fundamental, but not sufficient…



(note)Learn these as you go through university. Technology, frameworks, tools and techniques all change rapidly.

2. <Skills Framework for the Information Age (SFIA)>



The skills context

IT professional capability comes from a combination of professional skills, behavioural skills and knowledge. Experience and qualifications validate that overall capability.

Professional skills. Business process improvement and Database design are just two examples of almost 100 fundamental professional IT skills defined by SFIA.

Behavioural skills. Most organisations recognise a set of behavioural skills. These vary considerably from one organisation to another.

Knowledge. Technologies, products, internal systems, services, processes, methods and even legislation are all examples of areas where IT professionals are required to have knowledge.

Experience and qualifications. These validate the individual's capability. Qualifications certify elements of skill or knowledge; experience gives practical demonstration of capability. The right sort of experience also acts as a powerful force for learning, thereby enhancing capability.

(notes)Focus first on professionalism.Been using the term as if there is a definition of what an IT professional is. Does this make sense when we know how many jobs there are that are considered IT. Who says what an IT professional is and what professionalism means in our environment?

3. <skills-team work>As one tutor phrased it: we do lots of human-computer interfacing, but human-human interfacing is just as important.

4.<computing teams>

-Team type(e.g.:Development; operations; QA; support; security; …)

-Team topologies(e.g.:Stream aligned; enabling; platform; …)

-Team roles(e.g.:Analyst; programmer; architect; tester; …)

-Team functioning (e.g.: pair programming?)

5.<what is a team>

•"A group of people with a full set of complementary skills required to complete a task, job, or project." http://www.businessdictionary.com/definition/team.html

•"A group is an intact social system, complete with boundaries, interdependence for some shared purpose, and differentiated member roles" [Hackman et al]

•"A **team** is a group of individuals working together to achieve a goal.

A group does not necessarily constitute a team. Teams normally have members with complementary skills and generate synergy through a coordinated effort which allows each member to maximize their strengths and minimize their weaknesses."

https://en.wikipedia.org/wiki/Team

6. <Team work-Diversity>

•What do we mean by diversity?

"the inclusion of different types of people … in a group or organization".

•We all have bias (explicit and implicit), assumptions, generalisations.

•Awareness of implicit bias.

•All teams are diverse and diversity matters.

7.<Team work-successful teams>

- Hackman identified three attributes of such groups
  - They satisfy internal and external clients
  - They develop capabilities to perform in the future
  - Members find meaning and satisfaction
- And then five factors that increase the chances for success:
  - A real team (shared task; clear membership; stability; …)
  - Compelling direction (SMART goals?)
  - Enabling Structure (size; internal structure; skills balance; …)
  - Supportive Context (reward; development; information; …)
  - Expert Coaching (support; mentoring; evaluation; …)

8.<Team work-IT professionals>

- Multi-disciplinary(e.g. business; IT; creative design; …)
- Multi-facet(e.g. analyst; architect; coder, tester; …)
- Collaborative(e.g. coder; coder; coder; …)
- Traditional plan-and-document structures
- Agile such as SCRUM or XP(eXtreme Programming eg pair programming)

9.<Team work- Successful Student teams>

- Sydney University Learning Centre
- Characteristics of groups that worked effectively:
  - equal contributions
  - full discussion of issues
  - member support

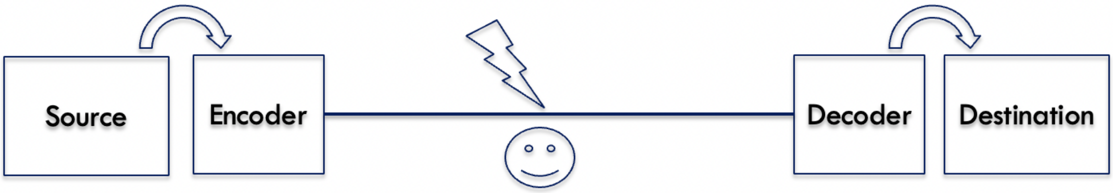
  ➡️High quality result & high level of member satisfaction
- Common problems that prevent groups working effectively:
  - problems with logistics
  - problems with allocation of tasks
  - coordination of member contributions
  - lack of commitment from some group members

  ➡️Quality of group product lower than individual product, & high level of stress and dissatisfaction
- Strategies for improving group dynamics
  - Setting up the group. Positive organisational systems such as drawing up a team constitution and open discussion in the first meeting of your group can help the development of a good dynamic.
  - Dealing with differences. In universities today, most groups are going to include people from different cultural backgrounds. Again, open discussion and tolerance are key factors for success here.
  - Dealing with negative behaviour such as aggression, blocking, controlling, freeloading and discounting.

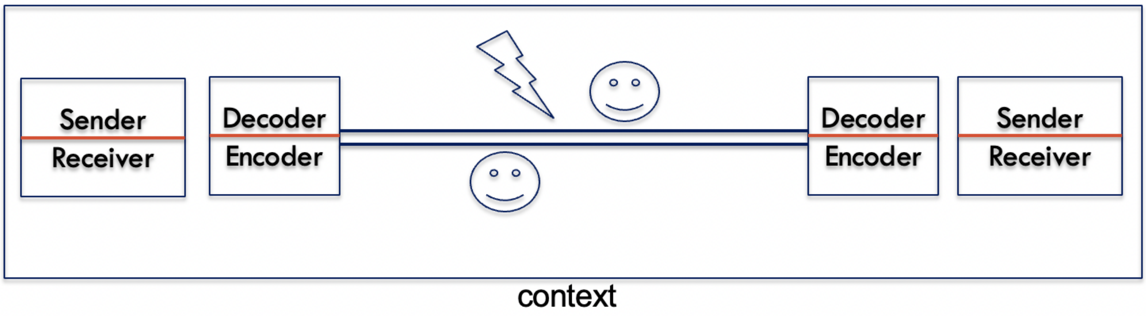
Week 3

1.<Communication>

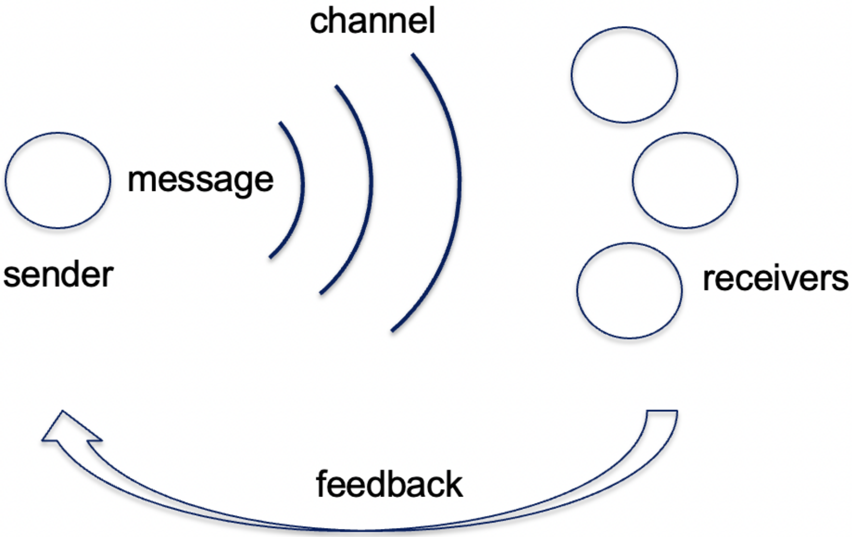
•Communication is “*a process by which information is exchanged between individuals through a common system of symbols, signs, or behavior*”
[from Merriam-Webster dictionary
https://www.merriam-webster.com/dictionary/communication]
2.<Transmission Models> Claude Shannon



3.<Interaction Model>



4.<Transaction Model >



5.<Communication-Forms>

•What differentiates these different forms of communication?

- •Different media (written, oral & slides, talking while looking at code in IDE, etc)
- •Different scales (1 page summary, 50 page report, 10 minutes talk, 3 hour conversation)
- •Different type of target (boss, subordinate, colleague, client, partner)
- •Different number of targets (one-to-one, broadcast to many)
- •Different patterns of interaction (frequent alteration of direction, always one-way, mostly one-way with questions)
- •Different purpose (convey facts, guide action, adjust attitudes)

6.<communication-your goal>

•After the communication, what change do you want in your targets (audience, readers, maybe even indirect influence)?

- •they know something
  - •e.g. they know what some tool can do
  - •e.g. they know *how to* use some feature
- •they will do something
  - •e.g. they will hire you
  - •e.g. they will approve a decision
  - •e.g. they follow a particular style when they write tests
  - •e.g. they will give you some information you need
- •they will have certain feelings/beliefs (affect)
  - •e.g. they will respect you

7.<communication-context>

•Know where your targets start from

- •What do they know?
- •What they expect?
- •What they value?

•To produce a desired outcome, your actions will be different depending on the starting point!

•Explain things they don’t know in relationship to what they do know

•Don’t use unexplained terms they don’t recognize

•Follow the style they expect (tone of language, dress, layout, etc.) unless you deliberately want them to notice the difference

- •e.g. to get attention to some point

8.<communication-respect>

•Your targets are busy.

•They are skilled in their own domain.

•They have their own goals.

- •Don’t waste their time:
- •be well-prepared

- •help them focus on what is most important
- •give them ways to exit quickly
- •connect what you want with their objectives or incentives

•Make use of their skills and knowledge:

- •don’t repeat things they know.
- •don’t do something they could do better.

9.<Communication – Common Mistakes in Writing>

•Poor organization (muddy thoughts)

- •instead aim to have one clear well-signposted message of each paragraph, and these should connect logically!

•Misuse of English idiom and grammar

•Tone too informal (or instead, too pompous) for the audience

•Digressions

- •Especially history of the writer’s thoughts or experiences, rather than the outcome and conclusion

•Lack of clear conclusion/recommendation

10.<Communication – Common Mistakes in Presenting>

•Monotone voice

•Overly soft voice

•Not looking at the audience

•Too much text on slides

•Too much “activity” on slides(animation, colours, fonts etc.)

•Not managing time well(especially: going over time!)

•Also: poor organization, tone not matched to the audience, poor English, digressions! (see earlier on written communication)

[Latex]

1.<What is Latex?(or Tex)>

•TeX – Typesetting program; LaTeX – formatting and “higher level” macros based on TeX

•Uses “plain” text rather than formatted text

•“Markup” tags are used to apply structure, style and formatting

•Compiled to produce an output file

<why?>

•Donald Knuth wrote the TeX language

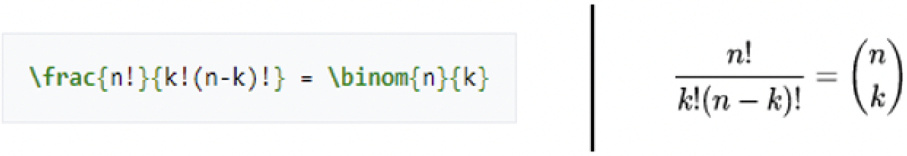
•“to allow anybody to produce high-quality books using minimal effort, and to provide a system that would give exactly the same results on all computers, at any point in time”

•Case study of software development including:

•Well documented changes

•Version control

- Collaboration - Rewards for finding bugs (starting at a hexadecimal dollar, US$2.56)



```
\frac{n!}{k!(n-k)!} = \binom{n}{k}
```

$$\frac{n!}{k!(n-k)!} = \binom{n}{k}$$

(https://en.wikibooks.org/wiki/LaTeX/Mathematics)

2.<Why using in info1111>

–Used extensively in Academic writing, but not really in industry.

–However…

  –Focuses writer on content + structure, and not presentation

  –Better at equations, layout, bibliographies, …

  –Encapsulates a number of key computing concepts

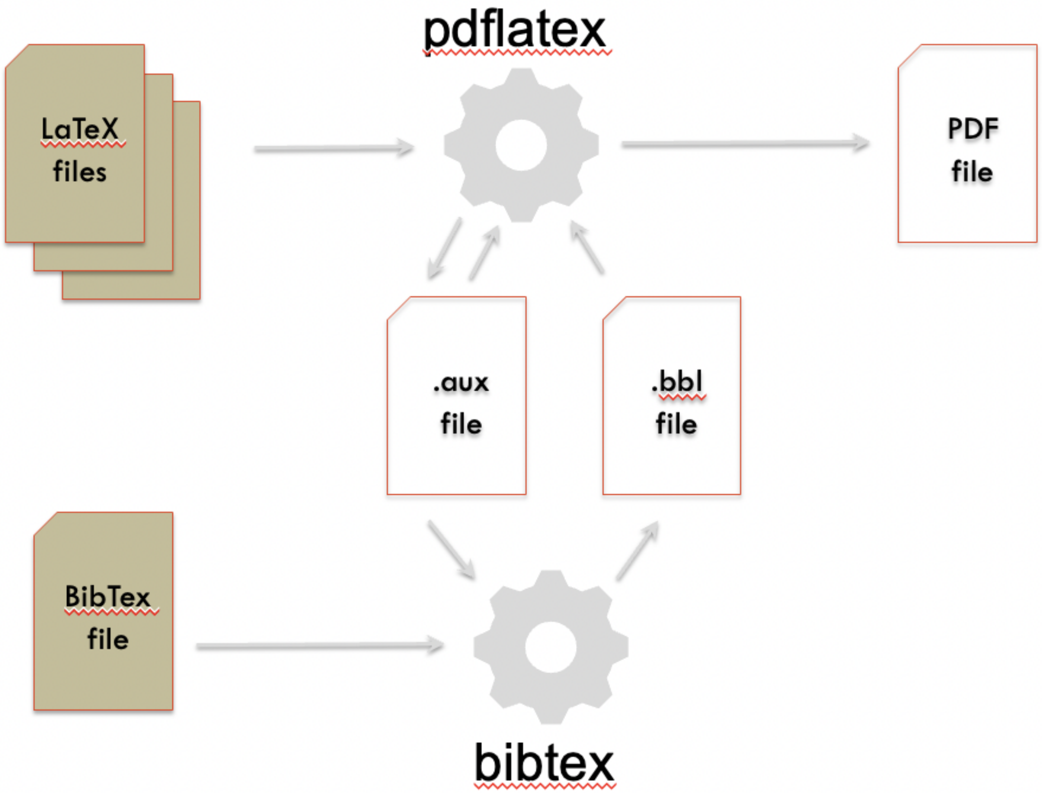
    - Compilation
    - Separation of concerns

–It is free

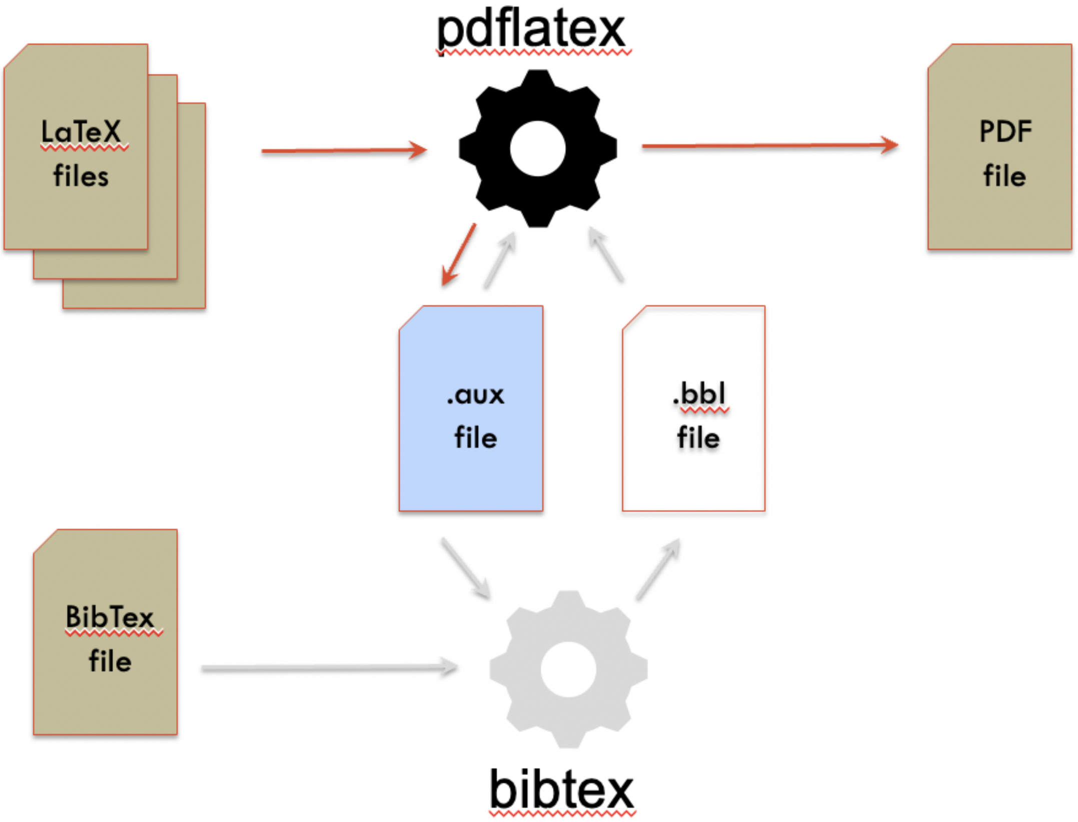
–We can use it to demonstrate tool use

3.<How it works>

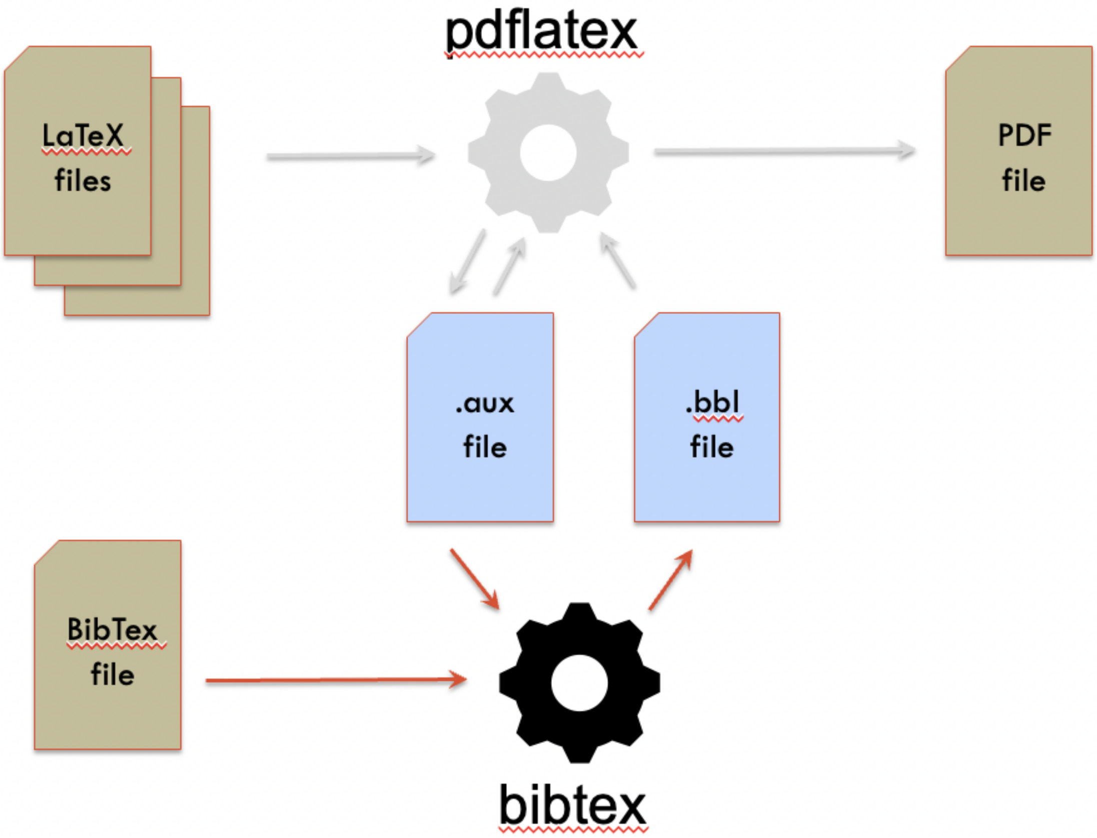
(1)



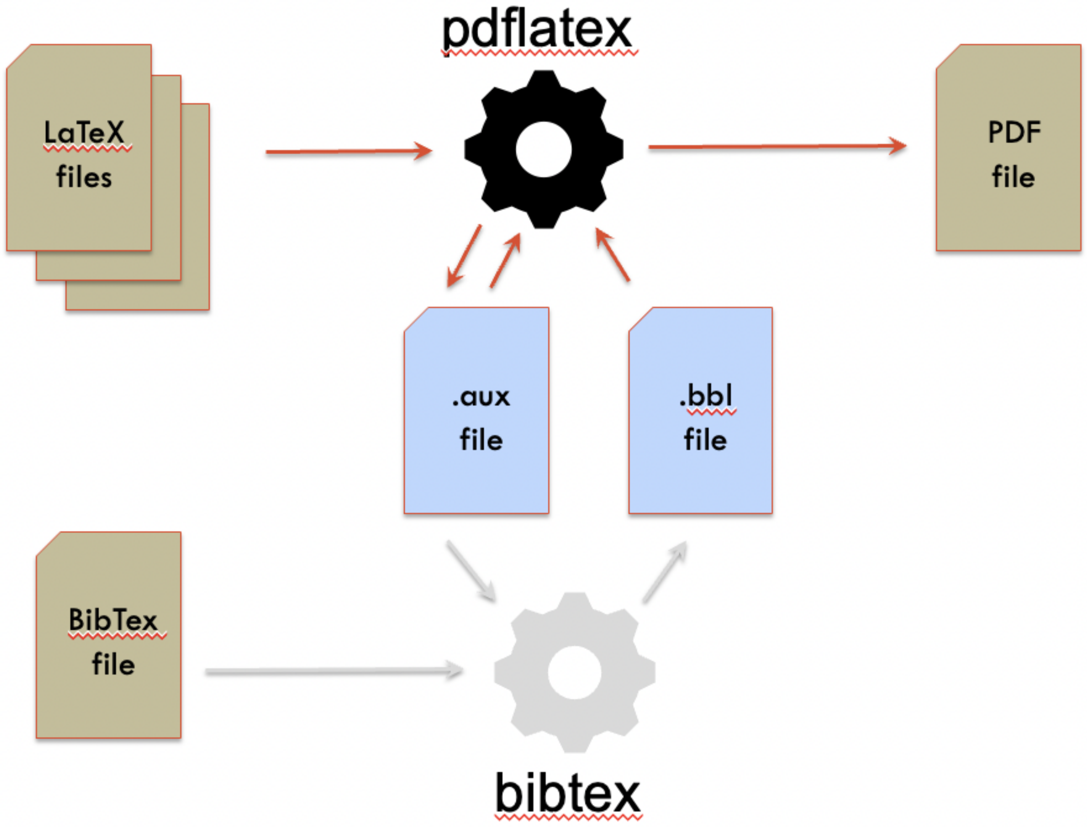
(2)


pdflatex
LaTeX files
PDF file
.aux file
.bbl file
BibTex file
bibtex

(3)


pdflatex
LaTeX files
PDF file
.aux file
.bbl file
BibTex file
bibtex

(4)

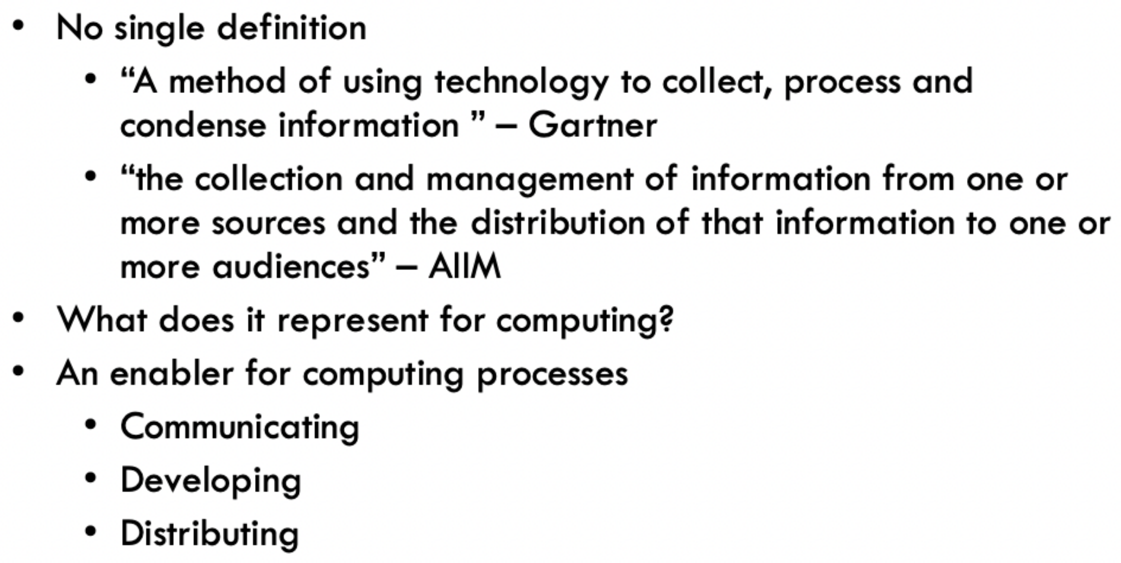


•Allows:

- •Focusing on content rather than presentation
- •Usage of pre-supplied style sheets
- •Effective indexing and cross-referencing

Week 4

1.<Information Storage & Management >

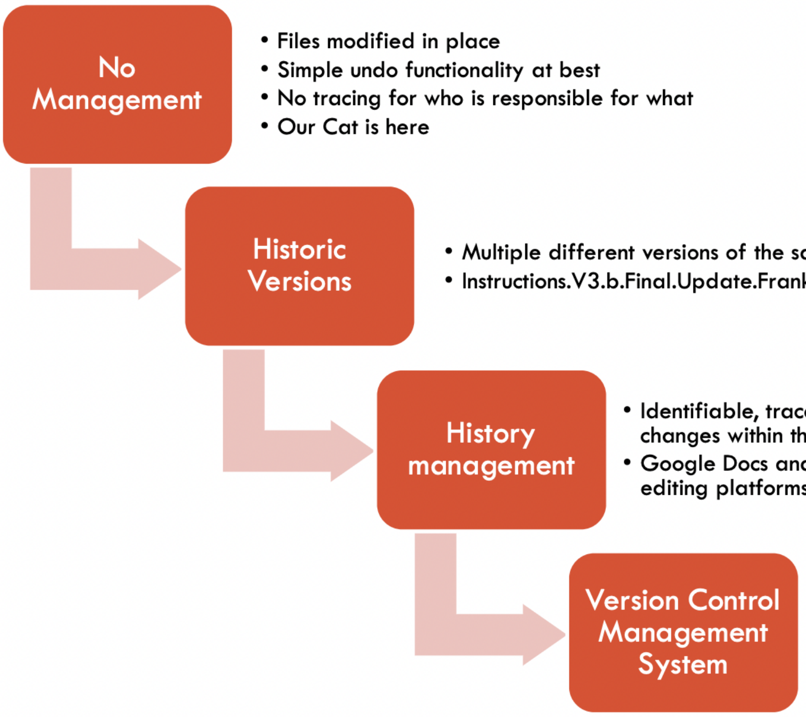


- No single definition
  - “A method of using technology to collect, process and condense information ” – Gartner
  - “the collection and management of information from one or more sources and the distribution of that information to one or more audiences” – AIIM
- What does it represent for computing?
- An enabler for computing processes
  - Communicating
  - Developing
  - Distributing

- What does it involve?
  - Creating information
  - Storing information
  - Sharing information
  - Changing information

- What information?
  - Non-computing business documents – contracts, inventories, plans, etc
  - Technical documentation
  - CODE

2.<An incomplete history of information management systems>



No Management

- Files modified in place
- Simple undo functionality at best
- No tracing for who is responsible for what
- Our Cat is here

Historic Versions

- Multiple different versions of the same information
- Instructions.V3.b.Final.Update.FranksVersion.docx

History management

- Identifiable, traceable, revertible changes within the document level
- Google Docs and other shared editing platforms are here

Version Control Management System

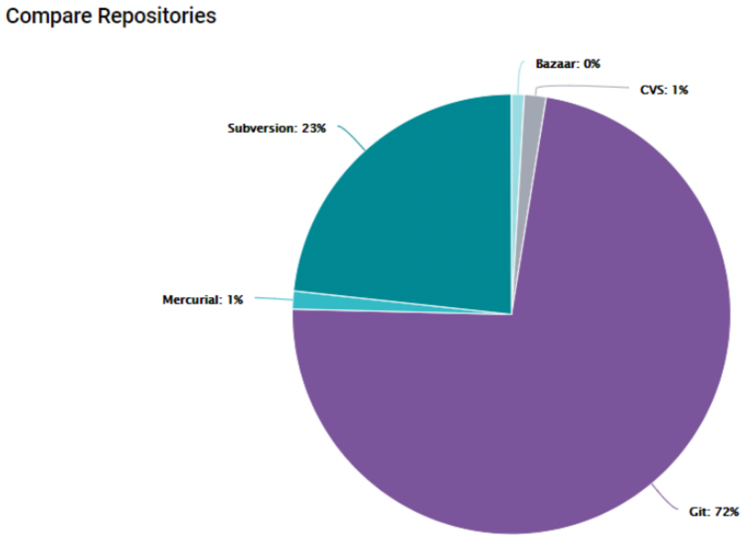
3.<Version Control>

- First, what are the problems we are trying to solve?
  - Large amounts of information
  - Modified by many people
  - In many different ways
  - Over a large amount of time.
  - Used in different ways

- Legacy
  - SCSS – Source Code Control System
  - RCS – Revision Control System
  - CVS – Concurrent Versions System
  - BitKeeper
  - Bazaar
  - SVN – Apache Subversion
- Current
  - Git
  - Mercurial

- What features do they provide? Some key words:
  - Stage
  - Commit
  - Merge
  - Branch
  - Push/Pull
  - Clone
  - Centralised vs Decentralised

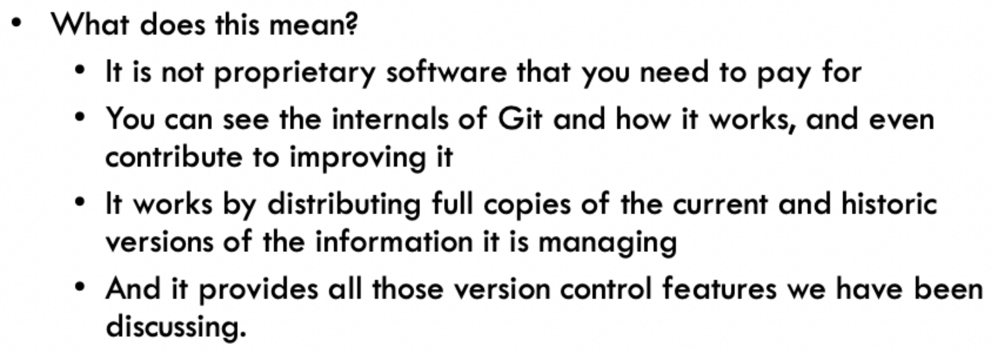
- At their core they all exist to solve the problems we have been talking about – enabling multiple people to collaboratively edit the same information simultaneously without stepping on each others' toes.



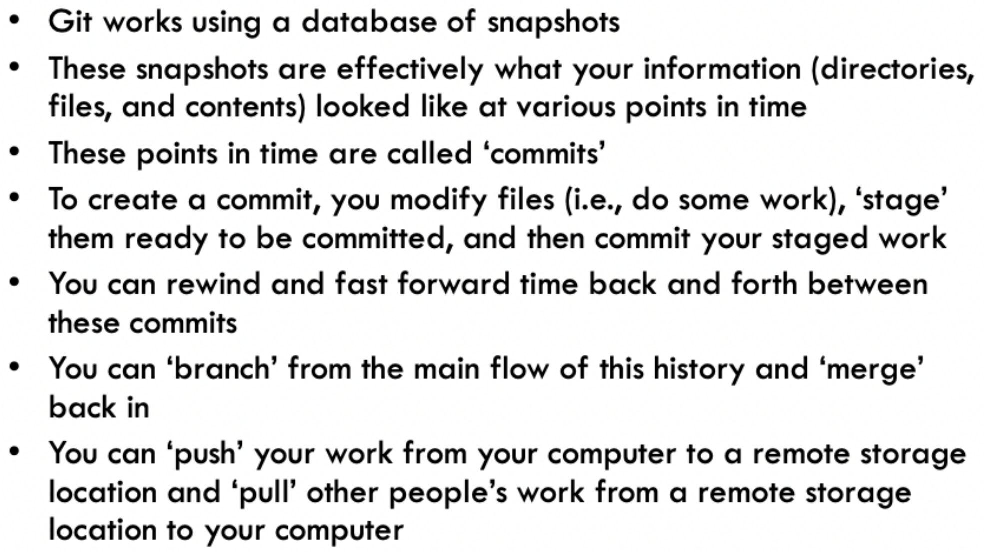
https://www.openhub.net/repositories/compare

4.<What is git?>

Git is a (1)Libre/Free (2)Open Source (3) Distributed (4) Version Control System (5) Also called a source code management tool.

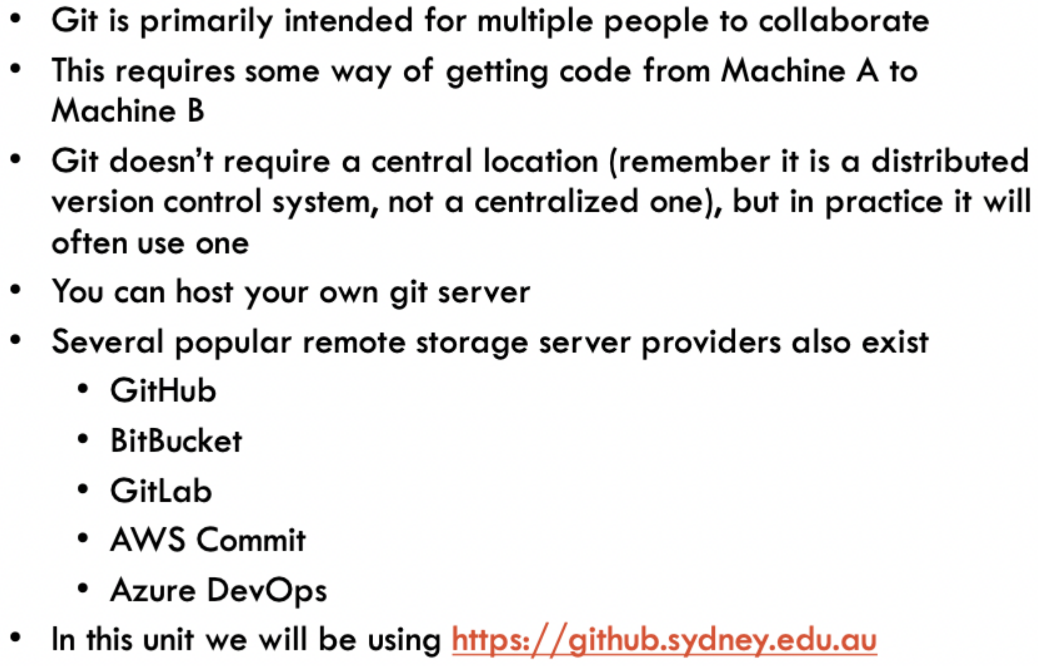


- What does this mean?
  - It is not proprietary software that you need to pay for
  - You can see the internals of Git and how it works, and even contribute to improving it
  - It works by distributing full copies of the current and historic versions of the information it is managing
  - And it provides all those version control features we have been discussing.



- Git works using a database of snapshots
- These snapshots are effectively what your information (directories, files, and contents) looked like at various points in time
- These points in time are called 'commits'
- To create a commit, you modify files (i.e., do some work), 'stage' them ready to be committed, and then commit your staged work
- You can rewind and fast forward time back and forth between these commits
- You can 'branch' from the main flow of this history and 'merge' back in
- You can 'push' your work from your computer to a remote storage location and 'pull' other people's work from a remote storage location to your computer

5.



- Git is primarily intended for multiple people to collaborate
- This requires some way of getting code from Machine A to Machine B
- Git doesn't require a central location (remember it is a distributed version control system, not a centralized one), but in practice it will often use one
- You can host your own git server
- Several popular remote storage server providers also exist
  - GitHub
  - BitBucket
  - GitLab
  - AWS Commit
  - Azure DevOps
- In this unit we will be using https://github.sydney.edu.au

Week 5

1. <Meaning of systems thinking>

•Broad level: The connection between solutions, systems that implement them and the society they operate in.

•More focussed: Components of a system, their interactions and interrelationships can be analysed individually to see how they *influence the functioning of the whole system.*

•… it's a way to view interactions in the wider world, a "way of thinking" or a "philosophy" for some.

<why is this different to other forms of thinking>

•Contrast to "traditional" systems analysis where a problem is broken into parts and studied individually

•In IT we need both …

  - •Computers and computational solutions currently focussed on "parts" and "interfaces"
  - •But this is a limited view ….
  - •Systems are highly dependent on "structure" or "relationships" among parts – how the parts are connected and this needs focus

(note)You cannot program a system – you need to program the components, decide on the hardware components etc.

•Systems are **complex** and **dynamic**

  - •Feedback loops (balancing, or reinforcing)
  - •Causality (not always clear)
  - •Emergent behaviour (e.g. adapting products for tasks that designers never intended – safety concerns)
  - •Behaviour over time

•Unpredictability of systems

<Importance>

•Unintended consequences

•Complex systems fail

•Tragedy of Commons

•Borneo sprayed with DDT to kill mosquitoes

  - •Reduced malaria by killing mosquitoes
  - •… but roofs started falling in

•Therac-25 (an example for almost everything …)

  - •"The equipment control task did not properly synchronize with the operator interface task, so that race conditions occurred if the operator changed the setup too quickly. This was missed during testing, since it took some practice before operators were able to work quickly enough to trigger this failure mode." https://en.wikipedia.org/wiki/Therac-25
  - •Solution exists in bigger system

•"Failure is almost never obvious until you're looking in the rearview mirror"

–System thinking: identify the interdependencies between elements of a system as feedback cycles and delays

- –What's perceived as an isolated problem can be part of an interconnected network of other issues
- –Identify positive and negative feedback cycles affecting a problem
- –Feedback cycles could take a long time
- –Problems could be the result of previously unrecognized feedback cycles
- –What the system is susceptible to can be leveraged to create change
- –What seems to worsen the situation may, in fact, lead to identify other solutions

2.<Application in IT industry>

•Example: Napster

•Also about decision making …

- •Software vs Hardware
  - •Telecommunications
  - •Cars
  - •"To develop the software, you need the hardware. To develop the hardware, you need the software."

•Instead of asking:

- –What data is useful for users?
- –What functionality can we provide?
- –How do we build different functionalities and visual interfaces to give them what we have?

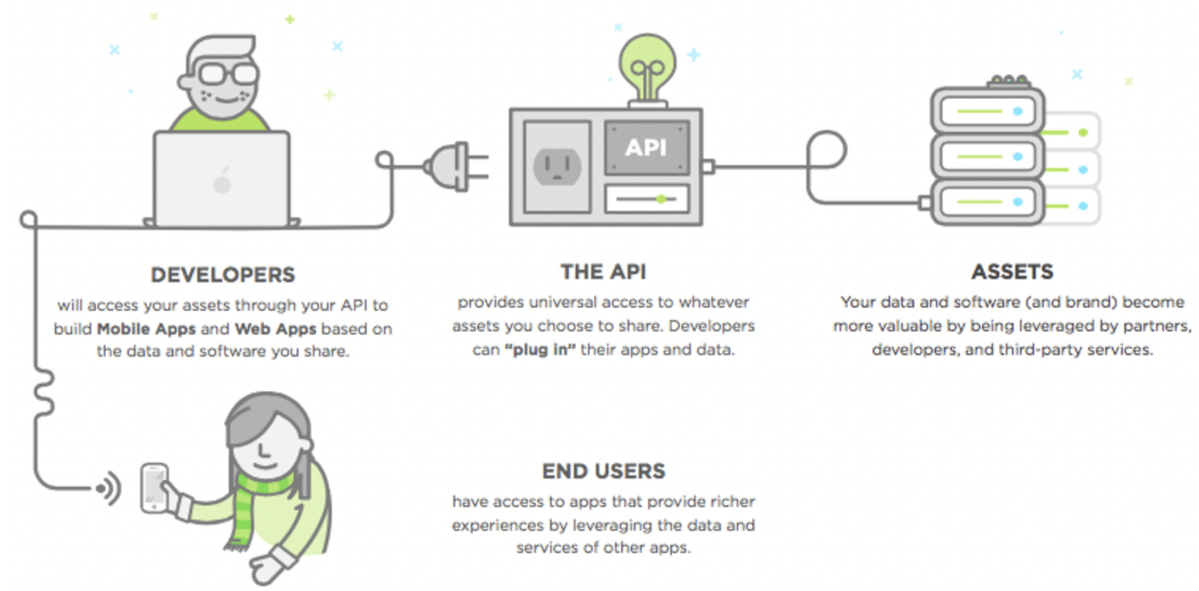
•and:

- –How are users actually interacting with existing systems?
- –How do our users want to engage with our services?
- –What are they aiming to achieve?
- –How can we make things as easy as possible for them?
- –How do we best give them what they need?

3.<API>Application Programming Interface

–"a set of clearly defined methods of communication among various components. A good API makes it easier to develop a computer program by providing all the building blocks, which are then put together by the programmer."

- –can describe how to connect a dataset or business process with a consumer application or another business process
- –create a seamless workflow for accessing a business' services
- –improve process efficiencies and authorisation movements across a business



DEVELOPERS

will access your assets through your API to build **Mobile Apps** and **Web Apps** based on the data and software you share.

THE API

provides universal access to whatever assets you choose to share. Developers can **"plug in"** their apps and data.

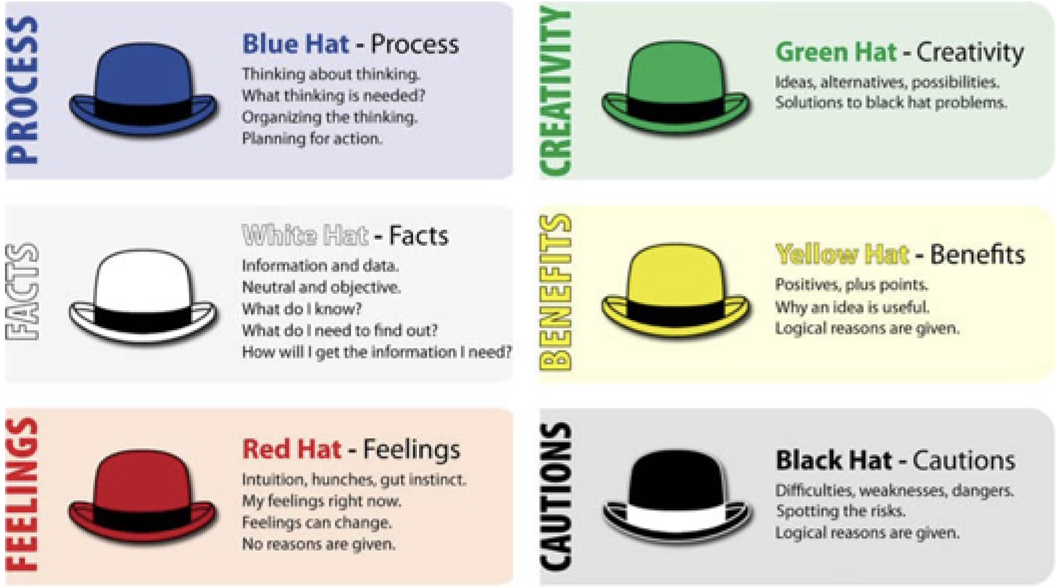
ASSETS

Your data and software (and brand) become more valuable by being leveraged by partners, developers, and third-party services.

END USERS

have access to apps that provide richer experiences by leveraging the data and services of other apps.

4.<Problem solving - in business>



PROCESS — **Blue Hat** - Process
Thinking about thinking.
What thinking is needed?
Organizing the thinking.
Planning for action.

CREATIVITY — **Green Hat** - Creativity
Ideas, alternatives, possibilities.
Solutions to black hat problems.

FACTS — **White Hat** - Facts
Information and data.
Neutral and objective.
What do I know?
What do I need to find out?
How will I get the information I need?

BENEFITS — **Yellow Hat** - Benefits
Positives, plus points.
Why an idea is useful.
Logical reasons are given.

FEELINGS — **Red Hat** - Feelings
Intuition, hunches, gut instinct.
My feelings right now.
Feelings can change.
No reasons are given.

CAUTIONS — **Black Hat** - Cautions
Difficulties, weaknesses, dangers.
Spotting the risks.
Logical reasons are given.

•The last two puzzles emphasise the point that before solving a problem, you need to understand what the problem is!

•In a professional IT environment this is often:

•Problem specification

•Requirement specification

•Understanding constraints

•etc

5.<Problem Solving Activity>

•Five volunteers – 1 problem solver, four agents

•Problem solver:

  •Put the agents in order

  •Rules:

- •You may only swap adjacent people
- •*(note: this is an example of having the solution constrained by the "implementation language")*

6.

–Problem solving is: Challenging; Rewarding; Creative

–Approaches: identifying possible alternatives

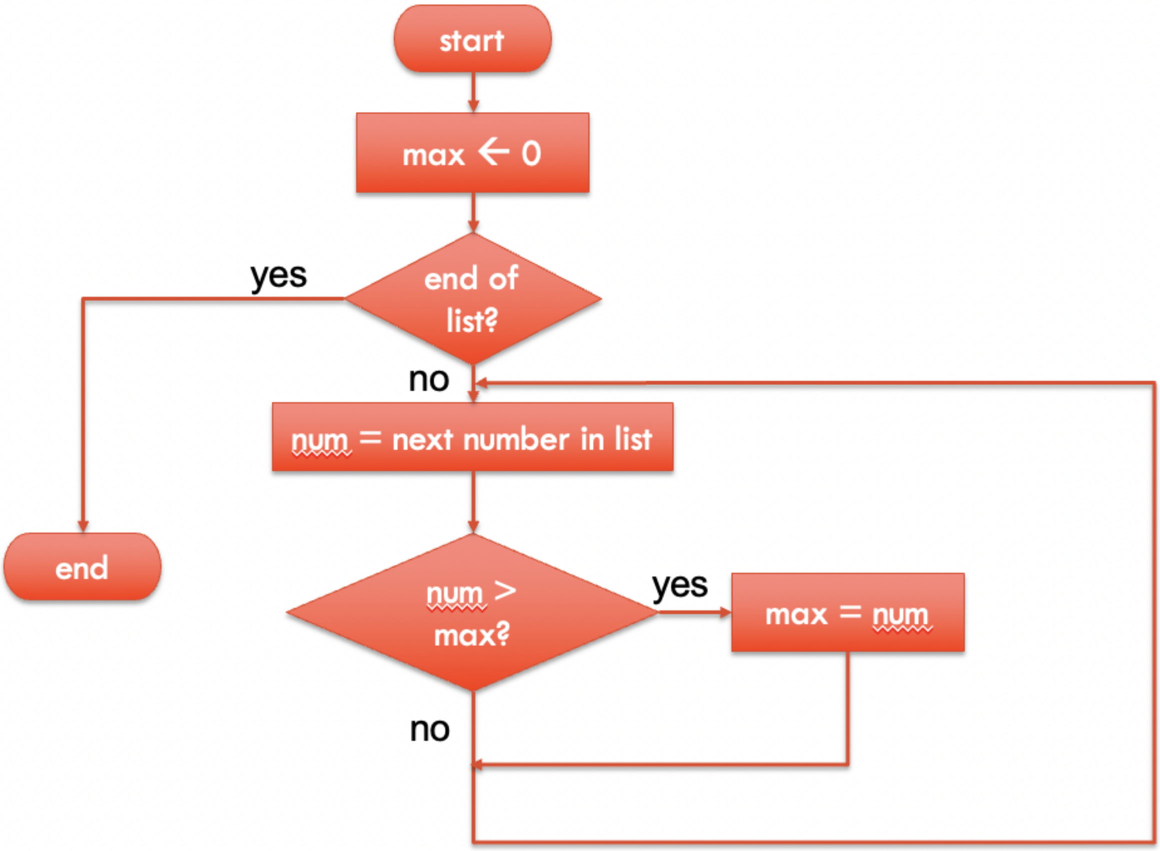
- –Brainstorming: forming; STORMING; norming; performing
  - •Word association; mind mapping; word banks; visual association; ask what if; exaggeration; think prepositionally; ask why; role storming; super powers; SWOT analysis; question assumptions

–Decomposing problems

- •Break the problem into parts (divide and conquer). But how?
  - –Functional? Procedural? Domain? Role?
- •Minimise coupling – the dependencies between components
- •Maximise cohesion – the focus of each component

7.–What is an algorithm? (vs a heuristic)Step by step instructions: exact; terminating.(sorting)

8. <Representing algorithms>



9.–P = Set of problems whose answers are easy to find

–NP = Set of problems where:

- –answers (if provided) can be checked easily
- –but not easy to solve (except by brute force)

Week 6

[Copyright]

1.<Intellectual Property in Technology>

> –(IP) The <u>ownership of ideas</u> and <u>control over the tangible or virtual representation of those ideas</u>. Use of another person's intellectual property may or may not involve royalty payments or permission, but should always include proper credit to the source.
>
> *The Free On-line Dictionary of Computing, © Denis Howe 2010*

–But then what is meant by “ownership” and “control”

–You get to benefit

–You get to decide

2.<History of Intellectual Property in Technology>

–14th C: Letters patent (letters = documented; patent = open/public)

> –Issued by monarch, etc to establish some right

–1624: Statute of Monopolies

> –Act of the Parliament of England, to address abuses of the “patent” system (e.g. QE-I issuing patents for salt, paper, soap, …)
>
> –Usually seen as the first *statutory* recognition of patent law.
>
> –Preserved patents for “novel inventions”

–1710: Statute of Anne

> –aka Copyright Act
>
> –1662: Licencing of the Press Act: gave the authority to print literary works to *The Stationer’s Company* (a printers guild). Led to censorship etc.
>
> –Copyright Act moved the authority to the authors of the work (for a period of 14/28 years).
>
> –Long title “*An Act for the Encouragement of Learning, by Vesting the Copies of Printed Books in the Authors or Purchasers of such Copies, during the Times therein mentioned.*”

3.<Q>If you take a video of a street performance then who owns the copyright in that video?

–The answer is not always obvious…

–For painted portraits the person commissioning the work is the first copyright owner

> •Under some circumstances the artist can limit the use of the work for other purposes.

–For commissioned photographs the photographer is the copyright owner

> •Unless the photographs are of a private or domestic nature, then the commissioning party owns the copyright.

–For films, the copyright owner is the person who made it.

> •The 'maker' is the person who undertook the arrangements necessary for the making of the film.
>
> •But for live performances, the performers might also be considers ‘makers’

–For commissioned films the copyright is owned by the commissioning party.

4.<who owns software>

–Software is protected as a literary work

–Usage is governed by licensing agreements.

–But this doesn't cover things like titles, images, etc.…

–Ownership determined by:

–Any agreement that is in place

–If created as an employee in the course of employment à the employer

–Otherwise à the creator of the software (generally the developer/s who wrote the code).

5.<General principles>

–Differing ways of protecting your ideas:

–Trade Secrets

–Copyrights

–Patents

–Trademarks

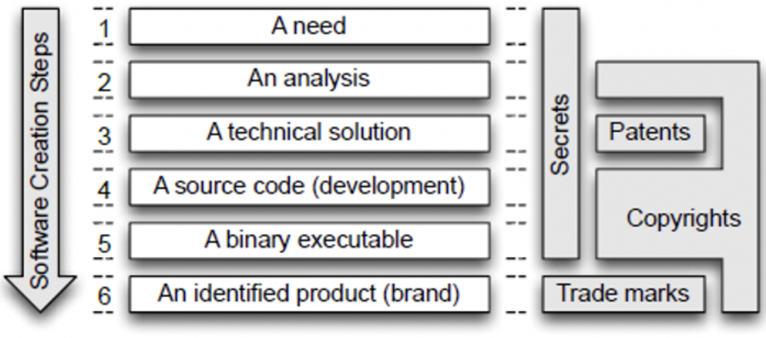


Figure 1

6.<Trade secrets>

–Protect your ideas by hiding them

–If something is not generally known, and you take steps to keep it secret, then it has protection under intellectual property law.

–But how do you hide software?

–If the creation process takes time…

–Then simply keep the idea secret and then be first to market!

–Someone else cannot use the idea unless they came up with it independently…

–So how do you keep it a secret?

–You must make *reasonable efforts* to maintain the secrecy.

•Inventories that identify material; confidentiality agreements and policies; physical and electronic security, etc.

–Only distribute executable and not source…

7.<Copyright>

–(with caveats) whoever writes the code owns the copyright

–An automatic legal right

•Varies depending upon whether the representation is “fixed” to a tangible medium.

•In US and Canada

–requires that works are "fixed in a tangible medium of expression” in order to be protected

•In France and Australia

–works need not be in a particular form to have copyright protection

–Whoever owns the copyright, has certain rights (which others do not have)

–Reproduce the software (both to physical media and with a computer)

–Publish the software

–Adapt the software

–Communicate the software to the public

–But there are a number of free use exceptions

–Fair dealing (research; criticism or review; parody or satire)

–Flexible dealing (for use in teaching)

–Educational exceptions (classroom demonstrations)

–Back-up copies

8.<Copyright over APIs>

–Consider the Google Maps API

–The “Application Programming Interface” defines the way in which a third party program interacts with a system.

–But would it be acceptable for me to:

–create a different mapping system, using my own maps;

–and then add an interface the same as the Google API?

9.<Patents>

–Protects a technical solution / invention

–A reward for investing in the development of the invention?

–The invention must be new, inventive, and useful

–Algorithms and abstract concepts cannot be patented

–Essentially gives an exclusive monopoly on an invention

–You can stop others using your invention.

–You can license others usage.

–Requires the “invention” to be described to the patent office (IP Australia)

–The information then becomes public!

10.<Trademarks>

–Protects a specific and distinctive “brand”

–It can be anything that represent your market position: a word or phrase, a logo or picture, a sound or smell, or some combination.

–It must be actively used or it can rescinded

–It is distinct from the company name and the domain name!

–Owning one does not protect the other!

–Similarly to copyright – you don’t need to register a trademark

–But it helps!

example for trademarks:

11.<Licences>

–Once you establish ownership (of the IP, not the software), how do you allow others to use (and benefit) from it?

–Licence agreements!

–Proprietary

•EULA : End-User Licence Agreement (“licenced not sold”!)

–FOSS (Free and Open Source Software)

–Variation in the rights that are granted:

–Right to use / copy / modify / distribute / sublicence

–But also the liabilities that are accepted!

12.<Open Source>

–Software where the source code is made available!

–Does not mean that the “owners” do not retain rights over the IP!

–Various licencing models

–Copyright:

•I own the software. You can use it but not redistribute

–Copyleft (Protective/Share-alike): e.g. GNU

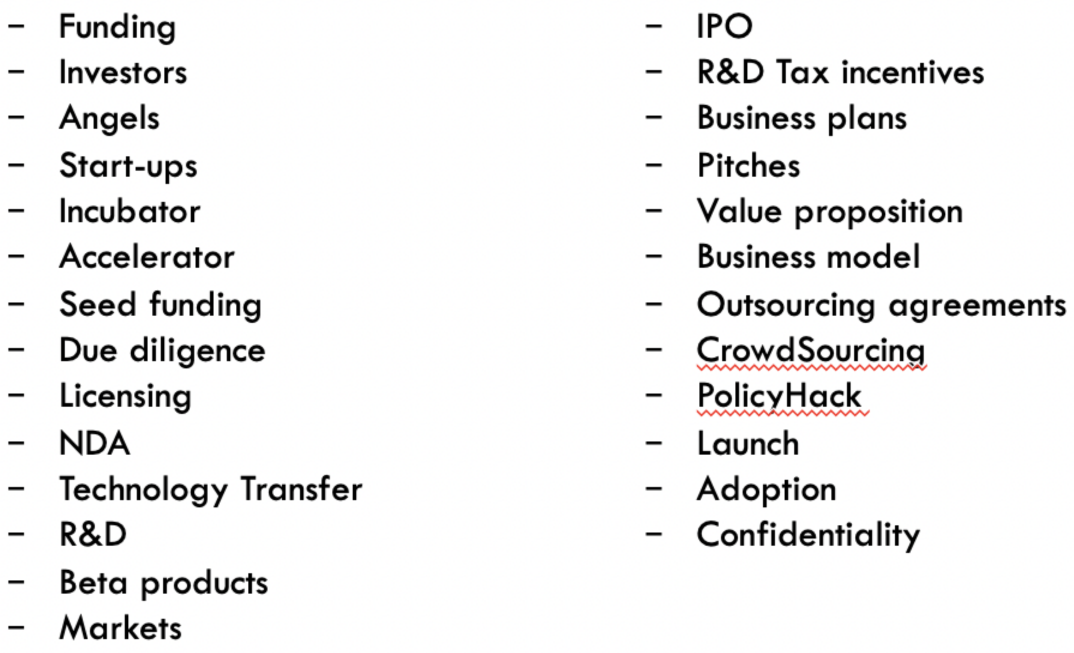
•I own the software. You can use it and redistribute it (for free), but the same rights must be preserved.

–Copycentre (permissive): e.g. BSD

•I own the software. You can use it and redistribute it (for free, or charged), you must acknowledge this software but can change the rights.

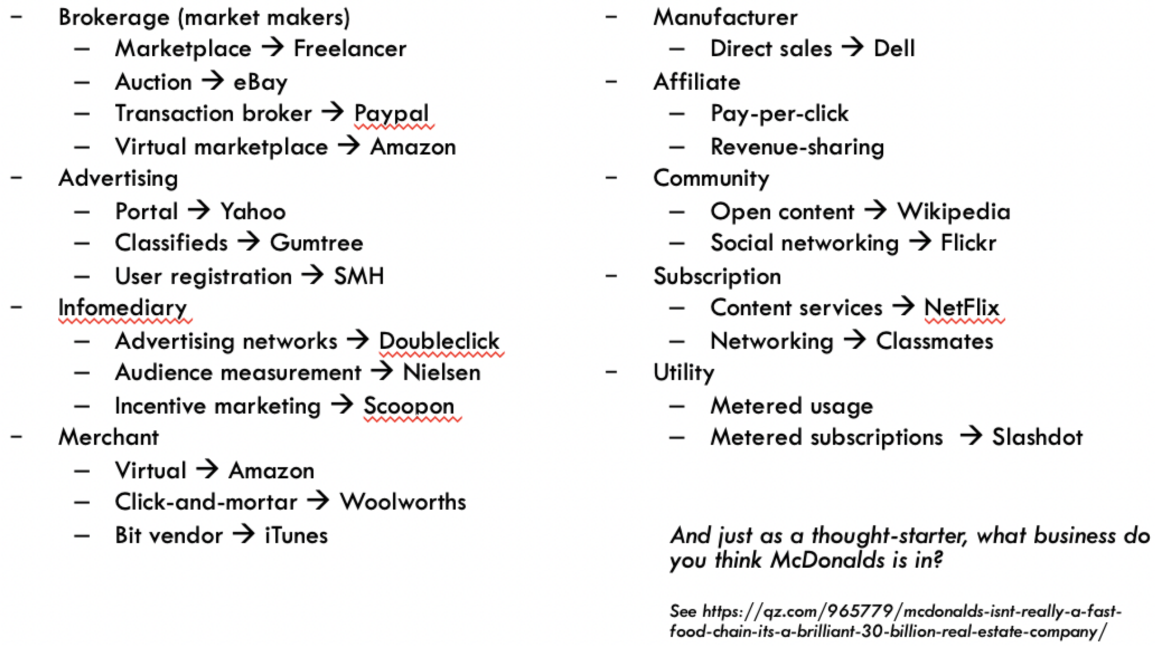
–Creative Commons

13.<Commercialisation>



- Funding
- Investors
- Angels
- Start-ups
- Incubator
- Accelerator
- Seed funding
- Due diligence
- Licensing
- NDA
- Technology Transfer
- R&D
- Beta products
- Markets
- IPO
- R&D Tax incentives
- Business plans
- Pitches
- Value proposition
- Business model
- Outsourcing agreements
- CrowdSourcing
- PolicyHack
- Launch
- Adoption
- Confidentiality

<Business Models>



- Brokerage (market makers)
  - Marketplace → Freelancer
  - Auction → eBay
  - Transaction broker → Paypal
  - Virtual marketplace → Amazon
- Advertising
  - Portal → Yahoo
  - Classifieds → Gumtree
  - User registration → SMH
- Infomediary
  - Advertising networks → Doubleclick
  - Audience measurement → Nielsen
  - Incentive marketing → Scoopon
- Merchant
  - Virtual → Amazon
  - Click-and-mortar → Woolworths
  - Bit vendor → iTunes
- Manufacturer
  - Direct sales → Dell
- Affiliate
  - Pay-per-click
  - Revenue-sharing
- Community
  - Open content → Wikipedia
  - Social networking → Flickr
- Subscription
  - Content services → NetFlix
  - Networking → Classmates
- Utility
  - Metered usage
  - Metered subscriptions → Slashdot

And just as a thought-starter, what business do you think McDonalds is in?

See https://qz.com/965779/mcdonalds-isnt-really-a-fast-food-chain-its-a-brilliant-30-billion-real-estate-company/

[Ethics]

14.Ethics is a branch of philosophy. Whole courses devoted to it.

•Ethics vs Morals?

•Morals: Principles of right and wrong that guide personal behaviour – your personal compass. Internal.

•Ethics: Rules of conduct accepted within a social context. External.

15.<Theory:Terminology>

•How do undertake ethical reasoning? Frameworks for making judgements…

•Ethical pluralism

•Teleological ethics

•Deontological ethics

•Egoism

•Utilitarianism

•Contractarianism

16.<Teleology>

•“State of the World”

•Egoism

•Value of a state is based on your individual situation

•… but think about flow-on effects

•E.g. how other people will respond, and how that will in turn effect you

•Utilitarianism

•Value of a state is based on total situation of all people

•“Greatest good of the greatest number”

•Who is included? How are their situations weighted?

    •E.g. Tradeoff between a few people suffering much, vs many people suffering a little

17.<Deontology>

•Decide on actions based on duty

    •rather than on determining the consequences in the particular case

•There are many duties, and they often conflict!

•Ross' prima facie duties:

    •Fidelity (promise-keeping, truthfulness)

    •Reparation (recompense for previous wrongs)

    •Gratitude (thankfulness for previous services)

    •Justice (happiness should reflect merit)

    •Beneficience (help others)

    •Non-maleficience (don't hurt others)

    •Self-improvement

18.<Professional frameworks>

•Most professional associations have "frameworks" guiding professional conduct

•These typically include consideration of ethical conduct as it applies in professional activities

    •Linking possible outcomes to duties

    •Guidance in setting priorities

    •Penalties?

•ACS: Code of Ethics, and Code of Professional Conduct

•ACM / IEEE-CS: Code of Ethics and Professional Conduct

•ACS: Code of Professional Conduct

    •1.2.1. The Primacy of the Public Interest

    •1.2.2. The Enhancement of Quality of Life

    •1.2.3. Honesty

    •1.2.4. Competence

    •1.2.5. Professional Development

    •1.2.6. Professionalism

•Detailed guidelines

•ACS has a procedure to handle complaints about members actions

•ACM / IEEE-CS: Code of Ethics and Professional Conduct (International)

•ACM recently revised its code (2018)

    •Addressed "the significant advances in computing technology and the growing pervasiveness of computing in all aspects of society since it was last updated in 1992"

    •A computing professional should...

    •1.1 Contribute to society and to human well-being

    •1.2 Avoid harm.

•1.3 Be honest and trustworthy.

•1.4 Be fair and take action not to discriminate.

•1.5 Respect the work required to produce new ideas, inventions, creative works, and computing

artifacts.

•1.6 Respect privacy.

•1.7 Honor confidentiality.

•

•2.6 Perform work only in areas of competence.

•2.9 Design and implement systems that are robustly and usably secure.

•3.1 Ensure that the public good is the central concern during all professional computing work.

•3.7 Recognize and take special care of systems that become integrated into the infrastructure of society.

19.<Corporate Ethics>

•How much does a company's ethical approach affect your choice.

•Would you work for a tobacco company?

•What if you were paid far more?

•Corporate Social Responsibility

•an organization's responsibility for the impacts of its decisions and activities on society and the environment.

•Fourth Bottom Line

•Profit, people, planet and progress/ purpose/ principles

(note)**Fourth Bottom Line** is a concept extended from the Triple bottom line; instead of simply focusing on the 3 Ps: people, planet and profit, this concept involves extending to a fourth factor which not only has motivation for a business but also transcends to a humanistic value and beyond by factoring in terms such as "spirituality",[1][2] "ethics",[3] "purpose",[3] "culture", "compassion".[4]

20.<Liability>

•Criminal: breach of the criminal codes (e.g. Vault corp worm?)

•Intentional tort: sue for damages

•Strict liability

•Negligence

•Is Windows bug free? What has MS done to avoid problems? What is their testing approach?

•Fraud

•Negligent misrepresentation

•Deceptive trade practice

•Unfair competition

•Misrepresentation of what your systems can achieve?

•Breach of contract

•How good are you at judging what might be feasible?

•Malpractice

    •Judged against a professional standard!

21.<Social Responsibility>

•“C4G is about the use of computing as a platform for improving the human condition”.

•Socially relevant computing

•Digital humanitarianism