Chapter Zero
Course introduction 2017

CZ3003 Software System Analysis and Design

Please open file and follow along …

S00.Course Intro 2017

(always bring your portable computer to class)

Syllabus description for course

Large-scale software can be an off-the-shelf product for universal usage, or a system engineered to meet specific needs. Both typically are part of a larger multi-disciplinary system, and required to predictably interact with external environments while carrying out complex operations.

Software analysis and design are critical to the successful development of a quality system. This CS-only course covers analysis and design along with implementation and deployment to prepare graduates for developing large-scale software.

ADVANCED subject, leveraging content of prior courses including…

Computer architecture

High-low level design

1006

2002

2005

2006

Design patterns

boundary entity control

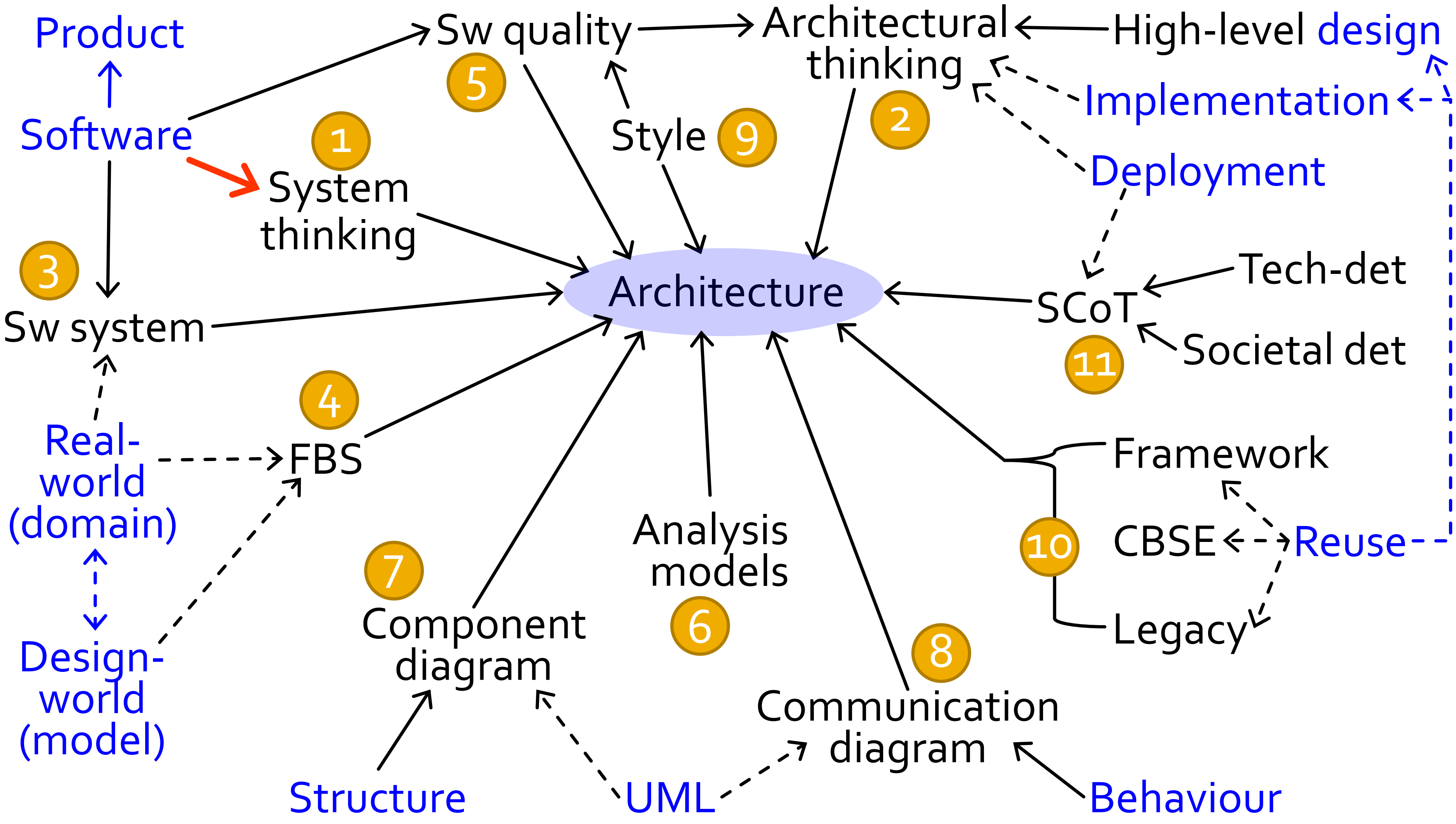
Content and its relations

Word-line Colour

Blue = prior know

Black = new

Red = deficiency



Product

Software

Sw quality

5

Architectural thinking

High-level design

Implementation

2

1

System thinking

Style

9

Deployment

3

Sw system

Architecture

SCoT

Tech-det

Societal det

11

4

Real-world (domain)

FBS

Framework

CBSE

Reuse

10

Legacy

Analysis models

6

7

Component diagram

Design-world (model)

8

Communication diagram

Structure

UML

Behaviour

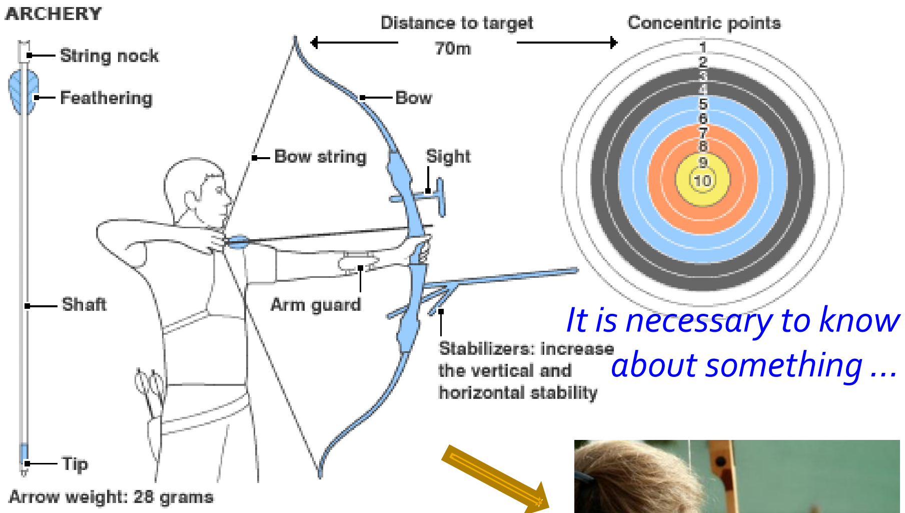
Content and relations (cont)

The focus of this course is sw *architecture*, and the primary skills are *crafting* and *modeling* it. The person practicing these skills is the architect. The inputs into architecture are the architect's *thinking*, sw *systems*, architecture *diagram*, sw *quality*, architectural *style*, *framework-CBSE-legacy*, and *social construction of technology*. Each of these are detailed in their own chapter in the textbook, along with identification of necessary prior knowledge.

This course is the next level of learning industrial sw development. Topics from these courses — *computer architecture*, *object-orientation*, *operating systems*, *software engineering* — are foundational for the content of this course. It is essential for this course that the student has a firm understanding of: 1) what is software; 2) what goes into sw products; 3) the SDLC stages of reqrs, design, implementation, and deployment; 4) UML; 5) what is sw structure and behaviour; 6) basic reuse in sw development; and, 7) who are stakeholders, and what are their responsibilities in sw development.

Learning outcomes are important

Traditional teaching is bound to what the student is supposed to *know*. Knowledge is pre-eminent, and it is established in the course as set of learning *objectives* for the student.



It is necessary to know about something …

… but only when you try something unassisted …

… can you really get to know it.

Improved teaching is engaged in what student is supposed to *be able to do*. Skill is pre-eminent, and it is established in the course as set of learning *outcomes* for the student.

What are your learning outcomes?

By end of course, in context of *sw*, student will be able to …

… *produce architecture* that is cogent and of specified quality

… *craft model* of architecture consistent with UML standards

… 1) *discern system* from product, and 2) *conduct SDLC activities* conducive to sw system dev

… *analyse sw system reqrs* w/system thinking & analysis models

… *imbue quality & style* into architecture of new sw system

… *employ framework/CBSE/legacy reuse* to develop sw system

… 1) *reason societal issues*, and 2) *close conflicts* between grps

Schedule

Week	Review (replacing lecture)	Tutorial	Lab
1 : 14 – 18 Aug	Intro ; Developing beyond sw product	–	–
2 : 21 – 25 Aug	Appreciating nuance of sw system ; TBL	Sys & arch thinking	–
3 : 28 Aug – 1 Sep	Sw architecture FBS Pt1 ; Pt2	System or product	Strategize P&A
4 : 4 – 8 Sep	Goodness to sw system ; Modeling Pt1	Derive cmpts	"
5 : 11 – 15 Sep	Modeling Pt2 UML Component ; TBL	Inprint quality	Do integration
6 : 18 – 22 Sep	Modeling Pt3 UML Communication ; TBL	Do analysis model	"
7 : 25 – 29 Sep	Putting it together (give demo; peer assess)	Make Cmpt diag	Start code
2 – 6 Oct	**Recess week**	–	–
8 : 9 – 13 Oct	Quality architecture with style; TBL	Make Comm diag	Start code
9 : 16 – 20 Oct	Reuse in sw system Pt1 ; TBL	Apply style	Write docs
10 : 23 – 27 Oct	Reuse in sw system Pt2 ; TBL	Model fmwk & CBSE	"
11 : 30 Oct – 3 Nov	Affecting sw arch: Tech & society; TBL	Model legacy	Prep demo
12 : 6 – 10 Nov	Foundational sw architecture 1st half & 2nd half	Apply SCoT	"
13 : 13 – 17 Nov	Walkthrough previous exam	–	Give demo

Schedule (cont)

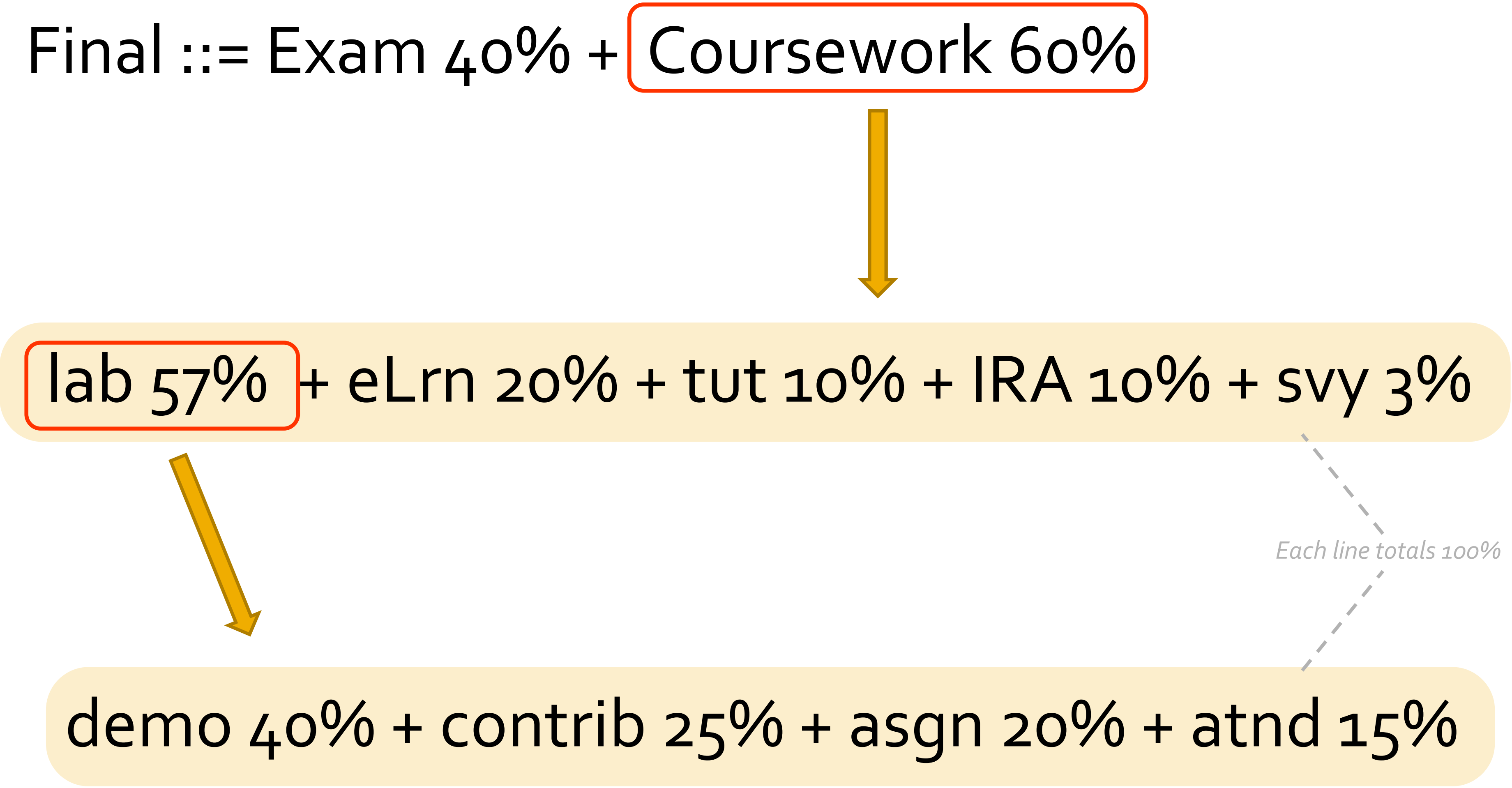
There are four official learning sessions, namely review (formally known as lecture), tutorial, lab, and eLearning.

The *review* is a combination of the previous semester's two one-hour sessions known as lectures into one two-hour session. In this two-hour session, the first hour always a topic review, and the second hour is either another topic review, or a workshop on the first topic. Reviews commence in week 1 as per your official timetable.

The *tutorial* and *lab* sessions are similar to those in previous semesters. Tutorials commence in "week 2", and labs in "week 3" on the even-odd week scheme as per your official timetable.

eLearning is an official activity (but not stipulated in your timetable) that opens one topic at a time at 1800 hours on the Friday preceding its applicable academic week. Eg., the eLearning opens on Friday 18 Aug for the week 21-25 Aug. The eLearning comprises discovery learning — drawing from the eBook, select papers, and videos — and confirmatory quiz. It is strongly recommended the student conduct it on that weekend and spend no more than two hours studying. eLearning commences in "week 0". *Contact teacher* via sms for any assistance involving the course content during the eLearning.

Assessment



Final ::= Exam 40% + Coursework 60%

lab 57% + eLrn 20% + tut 10% + IRA 10% + svy 3%

Each line totals 100%

demo 40% + contrib 25% + asgn 20% + atnd 15%

Assessment (cont)

All official learning sessions are covered in the coursework mark. Also, assessment is being shifted *away from team to the individual.*

In the first level breakdown of coursework, lab has the majority weighting because it entails the majority of the student's effort. eLearning is considered next in descending effort, as it comprises the discovery learning and confirmatory quiz in every week. Tutorial and review sessions are considered lower weighting as students have to take turns under the spotlight. Everyone can earn the survey mark; just participate. And having a firm grasp of your first two years of software engineering will have an impact, albeit indirect, on your assessment. Know your stuff: working with software, building sw products, every activity in the SDLC, modelling with UML, sw structure and behaviour, reuse in sw development, and, all the different stakeholders in sw development.

In the second breakdown, the demo and docs team efforts have larger share of marks. However, this is moderated by the individual contribution and attendance marks in *every lab session.*

Assessment (cont) – Must read!

Any and all quizzes in the official learning sessions are *effective dated;* there are no exceptions to enforcement of this!

Effective dated means there is a fixed period of time after the quiz is opened that students submissions of their responses are graded. Responses received after that time will <u>not</u> be graded. This is to ensure fairness of the grading, and facilitate keeping the continuous assessment to a tight schedule.

All quizzes are <u>open book</u> unless advised otherwise at the start of the quiz. Nevertheless, students are encouraged to "exercise" their memory of the topics, and do the quizzes as closed book.

For the quizzes in the tutorial and review sessions, you must be present in the classroom. This is to discourage cheating. The quiz for an eLearning session must be submitted within three days of the session's Fri opening.

Quiz grading timings will be announced through NTULearn.

Assessments will be posted on NTULearn. The assessment of the prior week will be posted on the week immediately following.

Bring your portable computing device to all quizzes to leverage the auto-marking. Any paper submissions of quiz responses will incur a 25% penalty.

Week seven assignment

This assesses the student's learning of content of the first *six* weeks; essentially it is a mid-term assessment. This is a *team* undertaking; no individual submissions. Assignment has two main tasks: A) post documents, and B) *give presentation*. Context of the assignment is *your lab project*.

Post. Each team is to provide at least one document for each of nine topics, as they apply to your assigned team's operations in the lab project as you understand them to be. The topics are: 1) system thinking, 2) architectural thinking, 3) sw system composition, 4) initial architecture structure with functions, 5) initial architecture behaviour, 6) imprinted sw quality, 7) analysis modelling, 8) *pending-final* architecture structure, and 9) *pending-final* architecture behaviour. Lab exec will advise the posting site.

Presentation. Each team will give a short verbal presentation of the *most noteworthy aspects* of its work so far. It shall be under 5 mins. The intention is to *impress the other teams* with how well your team has done in its project. This will be conducted in the classroom for the weekly review.

Marking of both tasks is by instructor and your peers. Marks by peers will be entered into LAMS rubric on the same day that marking is conducted. Teacher will advise details of LAMS in week six.

Week three-to-13 lab project

This assesses the student's learning of the *entire content* of the course. The student develops a *crisis management system* comprising team has a technical lead that collaborates with the other leads to integrate four teams, 911, CMO, EmergForce, and PMO; this is one Company. Each the four teams into one system. This is a *team* undertaking; no individual submissions. Project has two main tasks: A) post documents, and B) give presentation.

Final post. The mid-term assignment is a mid-course sanity check for your team. *Use it wisely*! Do not repeat the documents from the mid-term assignment. Instead, submit your *final* architecture (structure and behaviour) for your architecture and your best sales pitch models/video/etc to explain/justify/highlight your team's contribution to the system. Lab exec will advise the posting site. Teacher will advise deadline for submission.

Presentation. Each Company gives a multimedia presentation of its crisis management system, in a contiguous sequence of its four teams separately, followed by the system integration. Total time for the company's presentation is 90 mins. In week 12, Company will submit their teams' and integration individual presentation lengths and order to the *teacher for approval*, who then allocates day and time slots for each Company's prez in week 13.

LEARNING IN TRANSITION

Why TEL is important for learning

Instructional designs

Team-based learning, with conduct in two hour 'review'

Problem-based learning

eBook is your textbook, with papers and videos for more learning

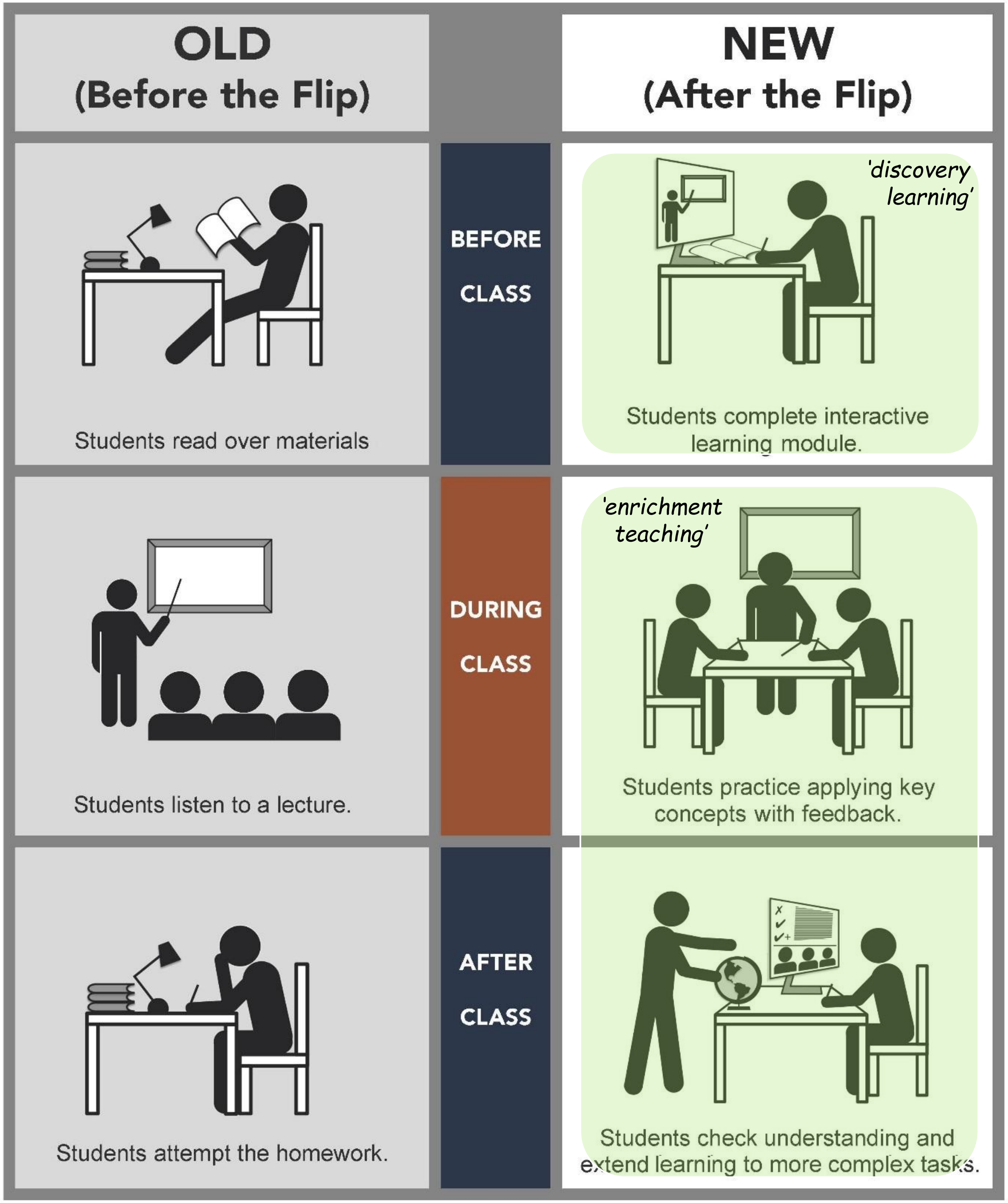
Read before lecture ⇨ imagery for recall

Why TEL is important for learning

Lectures are a thoroughly discredited instructional design. Unquestionably, they are incompatible with the modern student's familiar and automatic approach to learning; ever decreasing attendance and examination performances bear this out. From the success of the 2008-2011 blended learning project, NTU has finally established a campus-wide program to convert at least 50% of all courses in NTU to a more effective learning paradigm. Called *Technology Enabled Learning* (*TEL*), its mandate is to provide eLearning sessions brim-ming with *rich media* to support '*flipping the classroom*'.

Flipped classroom turns traditional learning and teaching on its head: the student engages in 'discovery learning' first, and the teacher engages in 'enrichment teaching' second. Consequently, the student actively learns the foundations of the material, and the teacher leads the student to deeper and more profound — albeit, more difficult — levels.

There are challenges for all parties in TEL. Teachers see their mandate as *'covering' the material* that is to be examined, and consequently *dumb down* the content. Students believe their mandate is to *earn marks*, and subsequently perceive the content as just a means of achieving this. Now, teachers, as content experts, must share the really crucial rather than



OLD (Before the Flip)		NEW (After the Flip)
Students read over materials	BEFORE CLASS	'discovery learning' Students complete interactive learning module.
Students listen to a lecture.	DURING CLASS	'enrichment teaching' Students practice applying key concepts with feedback.
Students attempt the homework.	AFTER CLASS	Students check understanding and extend learning to more complex tasks.

Why TEL … (cont)

superficial elements of the content. And students must now seize the topics and embrace their significance, thus learning the content not for marks but for itself. In other words, TEL obliges teachers and student to radically change their mindsets and give more of themselves.

To promote this sentiment in NTU of academic giving, I would start a new awareness program …

iGave

for momentous learning and teaching

Instructional designs

Flipped classroom maps to the official learning sessions as follows:

Discovery learning → eLearning session

Enrichment teaching → Review, tutorial, and lab sessions

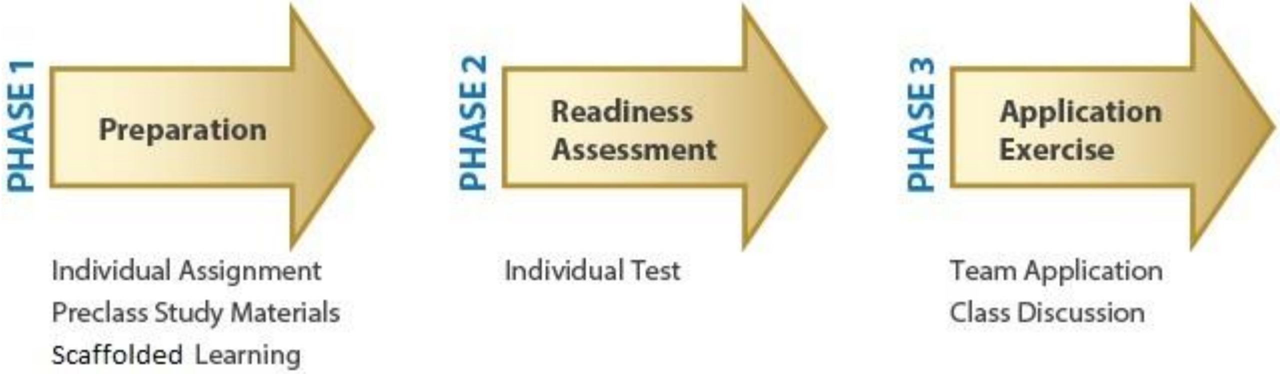
This course derives its learning and teaching activities from two innovative and widely-used (including other NTU schools) instructional designs: *team-based learning* (TBL), and *problem-based learning* (PBL). *TBL* is put to eLearning and review sessions, while *PBL* to tutorial and lab sessions.

TBL and PBL align to the learning sequence offered in the flipped classroom, plus they are complimentary and especially reinforcing in the recommended order: TBL is first, followed by PBL. TBL is designed for learning the 'material foundations', and its questions are direct, closed, and quantitative. PBL is designed for 'deep learning' beyond the foundations, and its questions are indirect, open, and qualitative.

For both, students are required to organize into teams. The students can belong to the same team (four to six members) for review and tutorial session; the lab team must be different (six to ten members).

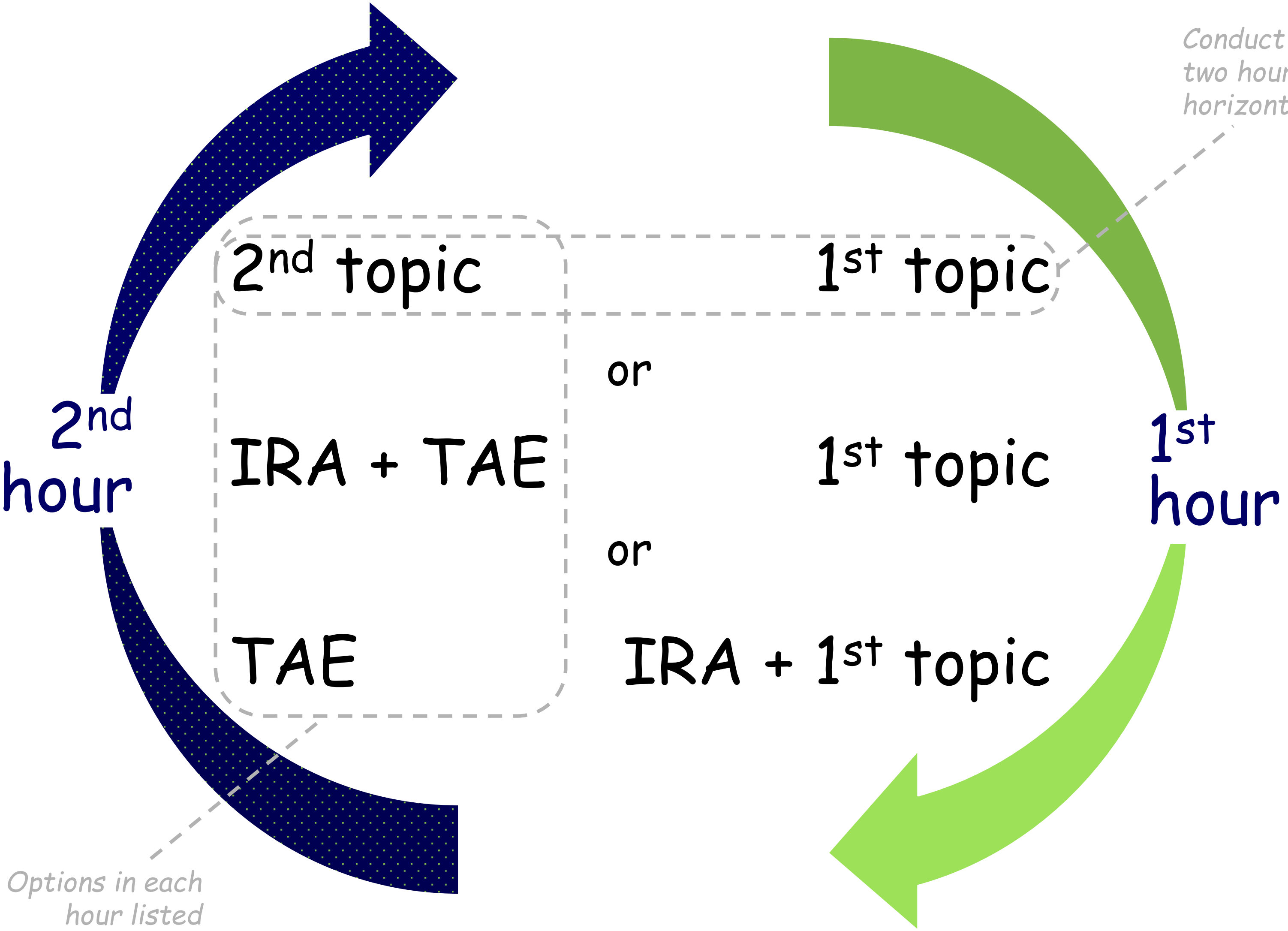
Team-based learning

TBL comprises three phases [see below]. Phase 1 is in the eLearning session, and Phases 2-3 are in the review session. In Phase 1, the student studies eBook/papers/videos, and answers a quiz. In Phase 2, the student takes an *Individual Readiness Assessment* (*IRA*), a quiz to determine his/her understanding of the foundation. In Phase 3, the student does a *Team Application Exercise* (*TAE*), a debate between teams of their own understandings of the material.



IRA is always before TAE, however, the starting time of the IRA is flexible. It can be before, during, or after the first topic [see figure next page]. The teacher will announce the IRA timing in the first minutes of the session. Regardless of the IRA timing, the TAE is held in the second hour and only if there is no second topic.

TBL phases in two-hour 'review'



Conduct of full two hours shown horizontally

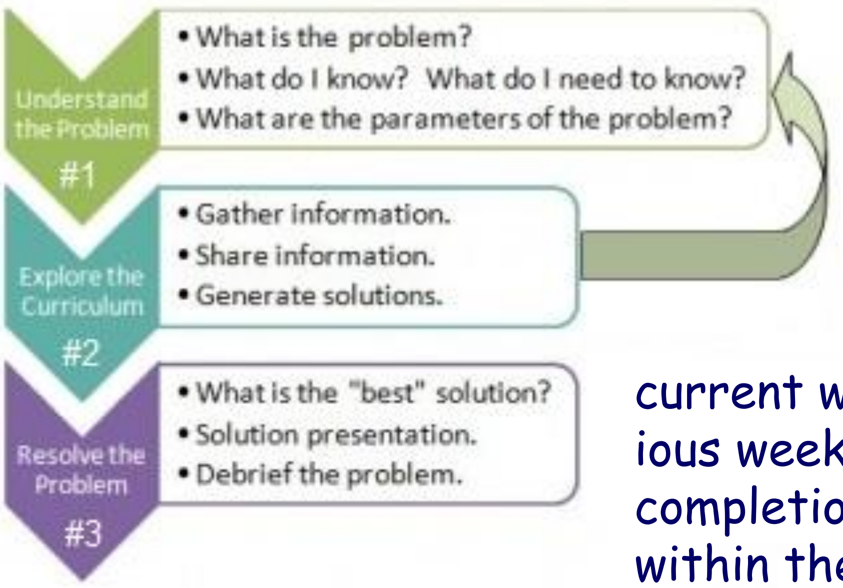
1st hour

2nd hour

2nd hour		1st hour
2nd topic		1st topic
	or	
IRA + TAE		1st topic
	or	
TAE		IRA + 1st topic

Options in each hour listed vertically

Problem-based learning



PBL for tutorial and lab sessions follow the same model [see left]. Only the time span of the assigned problem is different.

Tutorials have a new set of problems every week. The problems of current week relate to the topic from the previous week's. The problem is small enough to allow completion of all three stages in the PBL model within the one hour: 5 mins for question clarification, 25 mins for resolution, 5 mins to prepare prez, and 15 mins for all presentations. Each team's presentation should be four minutes duration.

The problem for the lab is *one* project spanning 10 weeks. Each of the five lab sessions focus on a segment of the total problem, chosen for its importance and cleverness in driving the overall lab effort forward to its eventual conclusion. So, each lab session is still PBL, except with the first two stages completed *before* coming to the lab session, and the third stage done *during* the lab.

Importance of your textbook/eBook

In this course, the instructor has written an eBook that is, *your textbook!* It is divided into ten chapters [see next slide] of which eight are on average 35 pages in length and contain roughly 5200 words. The material comprises the new learning concepts plus supporting knowledge on the big picture and prior knowledge for all the new concepts. Its purpose is to guide the learning and teaching of the curriculum of 3003. It is super important for your success this course.

It is not a compendium of data. It is a skills-based look at the subject, written from the perspective of a practioner with over 28 years in the business. This eBook gets right to the heart of the subject.

It follows the latest and best in *21st century learning and teaching.* Lecturing is out — it was discredited since the 50s; even your instructor viewed lectures as a colossal waste of time — so how are you going to become acquainted with the course content? Through this eBook!

The instructor knows your information input modality preference is *graphical* not textual. Accordingly, this eBook is *heavily illustrated*, with graphics that are optimized to convey exactly the meaning that the instructor is intending. The reviews leverage the book illustrations as the triggers for recalling primary parts of the content.

Download this eBook into your phone or tablet, carry it with you everywhere, and make it your go-to resource for anytime-anywhere learning. Its lessons are foundational and it will stay current and useful for a long time.

Ch1 Developing beyond software product

Ch2 Appreciating nuance of sw system

Ch3 Sw architecture FBS:
Real2Design object; Common function

Ch4 Ascribing goodness into sw system

Ch5 Modeling sw system: Misc analysis model;
UML Component; UML Communication

Ch6 [Left intentionally blank]

Ch7 Quality sw architecture with style

Ch8 Reuse in sw system: Framework & CBSE; Legacy

Ch9 Affecting sw architecture: Technology & society

Ch10 Prećis of foundation sw architecture

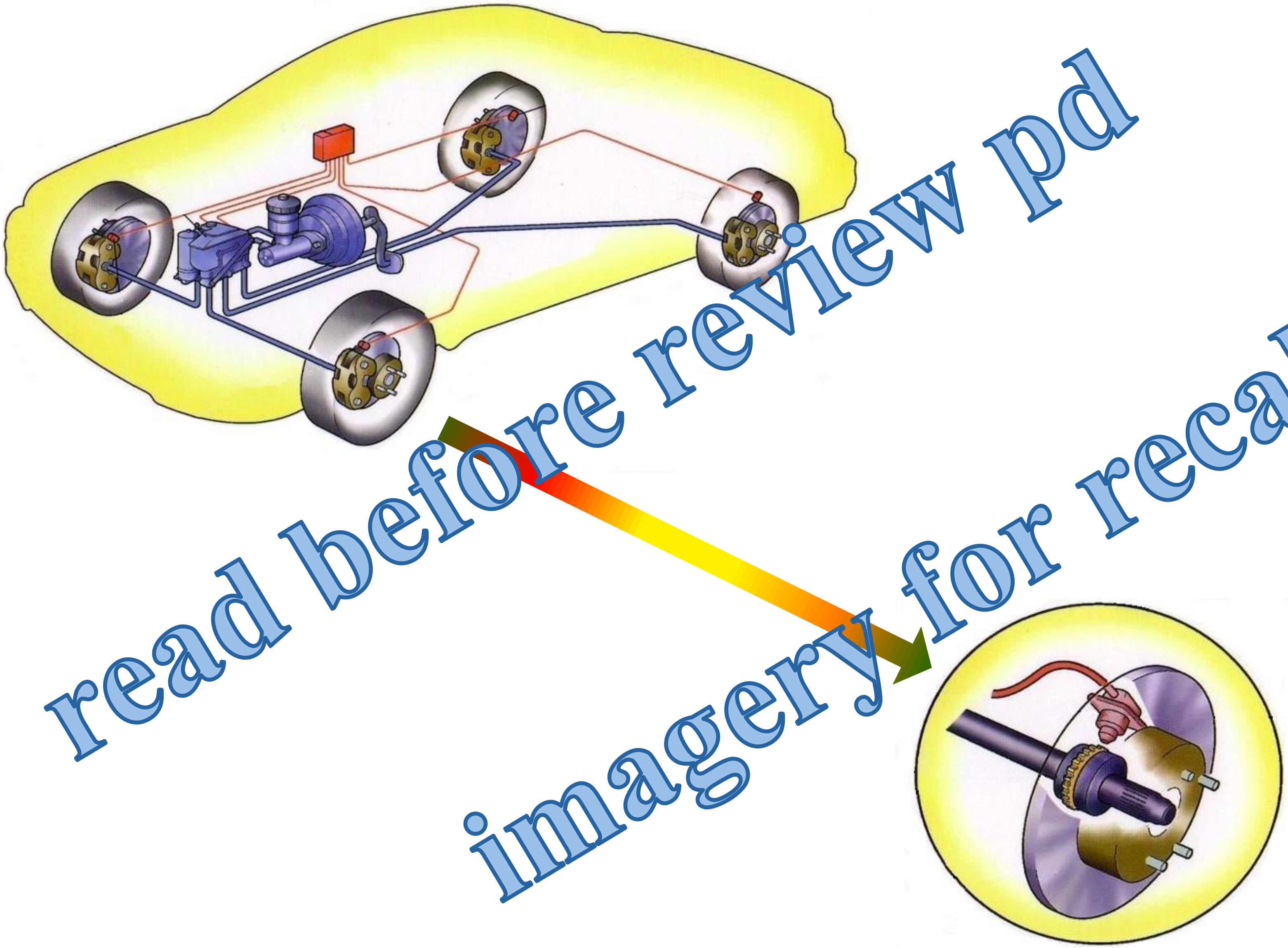
Resources for discovery learning

The eBook is considered to be the student's best resource for 'discovery learning'. It is comprehensively written, ie., all the learning items therein are written out in full. There are no bullet points or abridged content. There are many illustrations — nearly 400 — and many are accompanied with captions or explanatory notes highlighting important details in the figure [see next slide]. But it is not the only resource available for the student.

For those students who wish to delve further into the subject, there are over 30 published papers, briefs, and specifications to choose from [see next next slide].

Finally, for those students not so keen in reading to learn, there are over 20 videos to choose from, dealing with most of the topics' details. Also, these are encouraged to concentrate on the eBook's illustrations, as these will be primarily used in the review sessions.

Again, all students are *welcome to contact the teacher* by sms to seek assistance with any obstacle encountered in their discovery learning. Always seek *immediate clarification* of topic uncertainties!



IBM

What is Software Architecture?

From "The Rational Edge": This introduction to this relati
architecture is the first of a four part series on architectin
defining the discipline's key terms, and goes on to explo
contributes to the environment in which it is deployed.

Smashing Magazine

SMASHING MAGAZINE

Frameworks Round-Up: When To Use, How To Choose?

January 4th, 20

how-to-

InfoQ

Are You a Software Architect?

Posted by Simon Brown on Feb 0

Editor's note: The author of this artic
tutorial at QCon London coming up this Mar
article, entitled Software Architecture for De

The line between software development and software architect
people will tell you that it doesn't exist and that architecture is
design process undertaken by developers. Others will make o
that can only be crossed by lofty developers who believe you
abstractions and not get bogged down by those pesky implem

developersBOOK.COM
Learn from real world

Spring Framework Tutorial

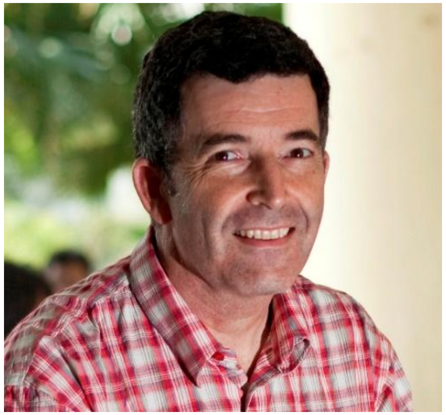
The Spring framework overview

standard in lightweight enterprise application framework. Spring is an
address the complexity of enterprise application
Spring framework is its layered
components you use while

papers

videos

Finally, your instructor



Kevin Anthony Jones

PhD (Learning), MEng (Sw)

Block N4, 02A-14

askajones@ntu.edu.sg

(O) 6790 4293 & (M) 97761716

"My door is always open to the inquisitive mind!"