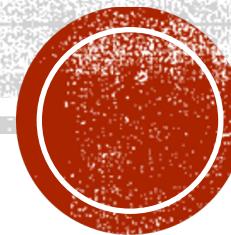


SOFTWARE ENGINEERING

Dr. Monika Gupta

monikagupta@iitj.ac.in

Researcher, IBM Research, India



WHY TO STUDY SOFTWARE ENGINEERING?

- Software are an integral part of almost every domain

Examples:

- System software
- Application software
- Engineering/scientific software
- Embedded software
- Web/Mobile applications

WHY TO STUDY SOFTWARE ENGINEERING?

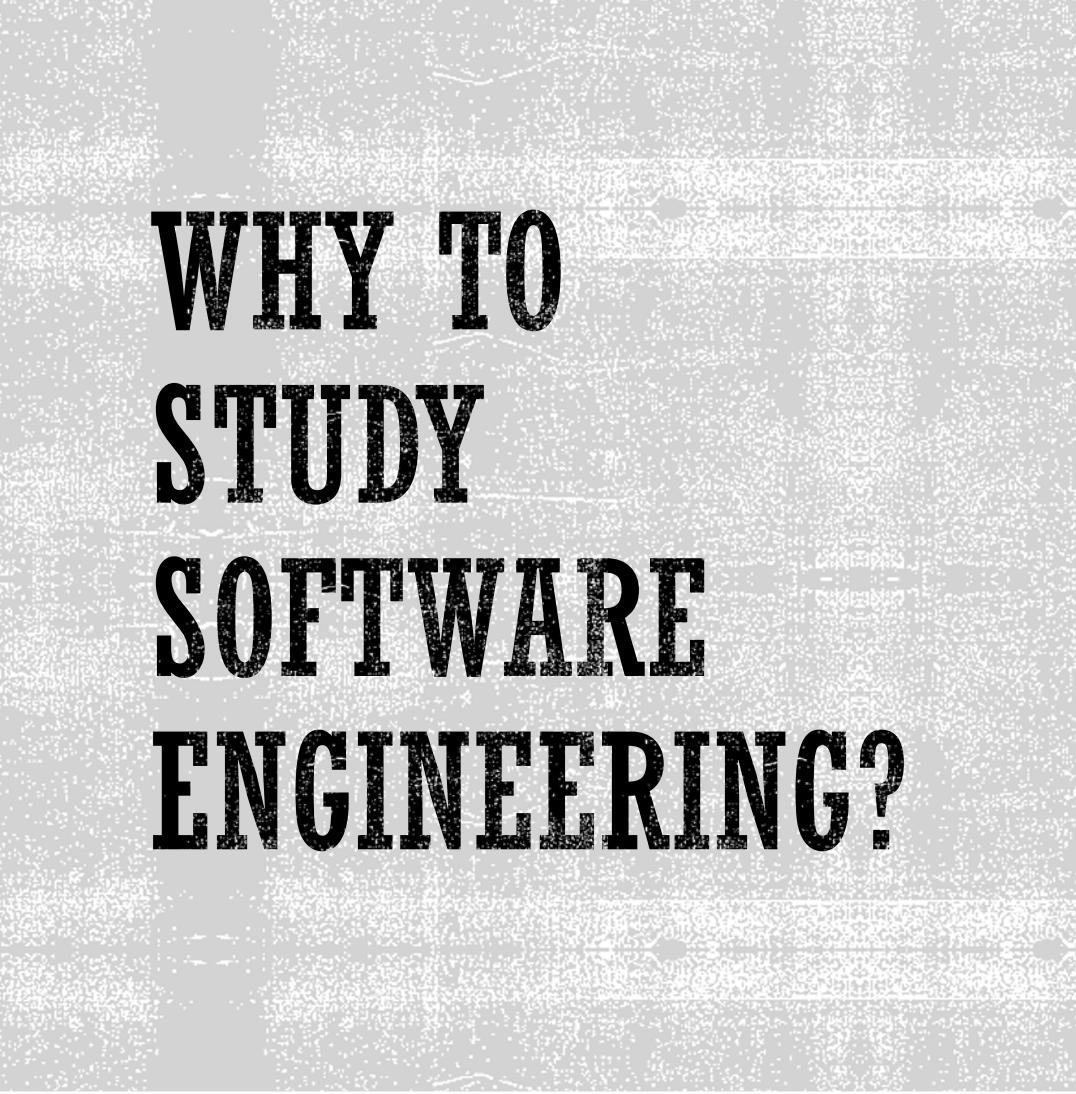
- One of the most complex things
 - Complex ≠ complicated

Suppose today is Tuesday, January 4



What day will be on November 29?

[Impossible to keep everything in your cognitive model]
[To answer, we need to bring the day names and the day numbers
into coordination, and for that we may need again a pen and paper]



WHY TO STUDY SOFTWARE ENGINEERING?

- Intangible
 - Not sure if it is working right
 - Anything we do to check if it is working right could be wrong itself

CATALOGUE OF CATASTROPHE

MODERN RESOLUTION FOR A		
	2011	2012
SUCCESSFUL	29%	27%
CHALLENGED	49%	56%
FAILED	22%	17%

The Modern Resolution (OnTime, OnBudget, with a satisfactory result) of all software projects in the CHAOS database. Please note that for the rest of this report CHAOS Resolution will refer to the CHAOS Resolution definition.

<https://www.infoq.com/articles/standards-modern-resolution>

CATALOGUE OF CATASTROPHE...

If you are interested in learning more debacles, visit this site
http://calleam.com/WTPF/?page_id=3



International Project Leadership Academy
Advancing leadership skills - Developing Project Management capabilities

HOME ABOUT WHY PROJECTS FAIL 101 COMMON CAUSES CATALOGUE OF CATASTROPHE

FEATURE ARTICLES CLASSIC MISTAKES TRAINING

Catalogue of Catastrophe

The following entries record some of the more noteworthy troubled projects from around the world (read our FAQ for more information).

[Jump to prior years 2015, 2014, 2013, 2012, 2011, 2010, 2009, 2008, 2007, 2006, Recent classics, Historical classics.](#)

For an analysis of the most common mistakes made in the projects listed below read the 'Classic Mistakes' page.

Latest troubled project report

Organization : US Department of Defense - USA
Project : Military aid
Outline : Forest pattern uniforms selected for desert type terrain
Date : Jun 2017 Cost : \$28M
[...read more](#)

Disasters 2016

1. Department of Public Works and Government Services - Canada - \$50M
[... read more](#)
2. City of Montreal - Canada - \$11M CAD
[... read more](#)
3. Aid Agencies - \$360M USD
[... read more](#)
4. Department for International Development (DfID) - UK - £285 million
[... read more](#)

Disasters 2015

1. Volkswagen - Germany - \$18B
[... read more](#)
2. Transportation Security Administration - USA - \$160M
[... read more](#)
3. Los Angeles Unified School District - USA - \$1.3B



FORMAL DEFINITION

Software Engineering :

The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software.



DISCLAIMER

- You may not follow any of the processes/tools taught in this course directly

GOOD NEWS

- Enable you to make engineering decisions necessary to adapt the covered templates/processes/tools to your needs, to your project's constraints

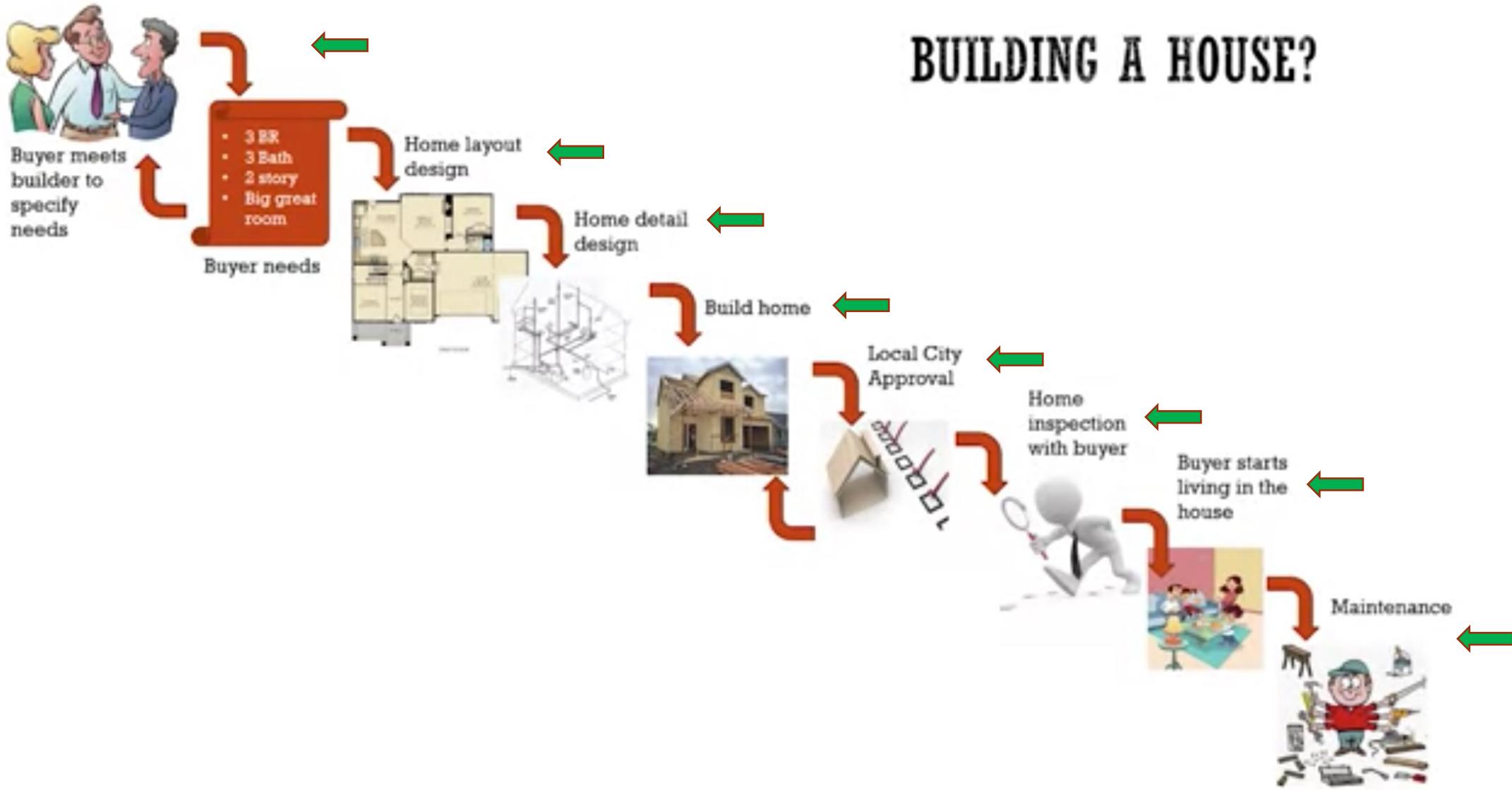




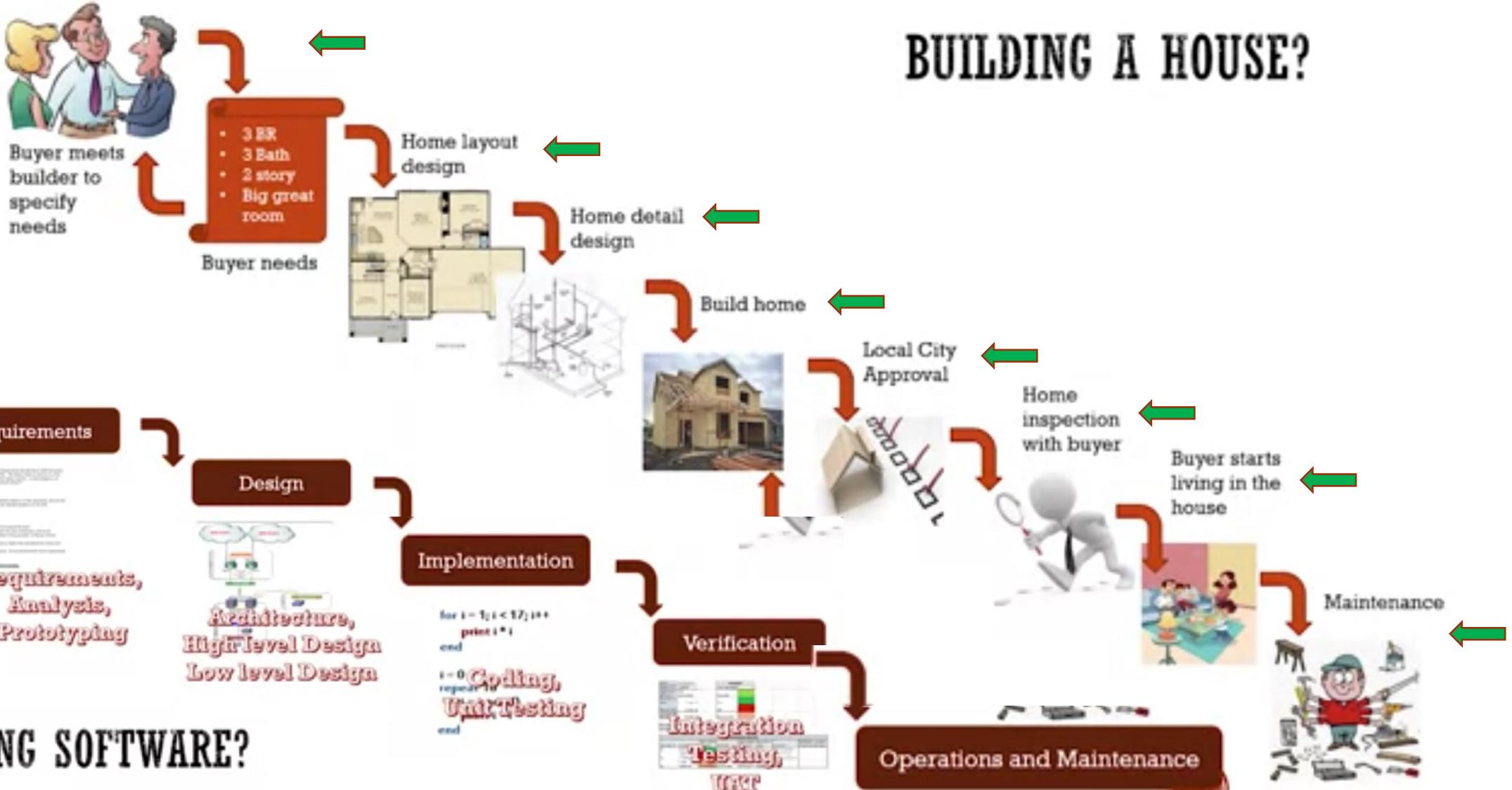
WHAT SOFTWARE DEVELOPMENT LOOKS LIKE?



BUILDING A HOUSE?



BUILDING A HOUSE?

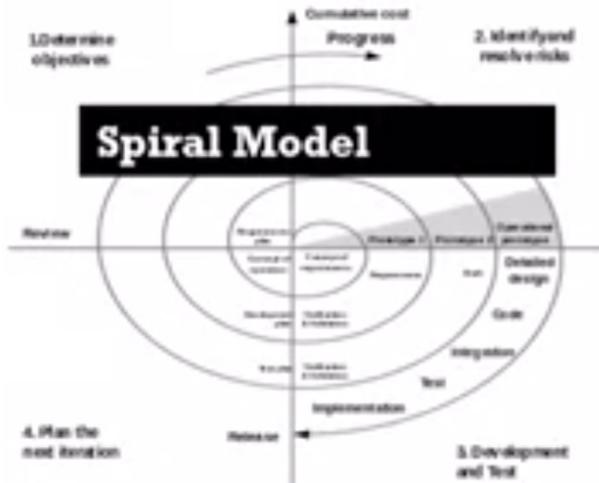
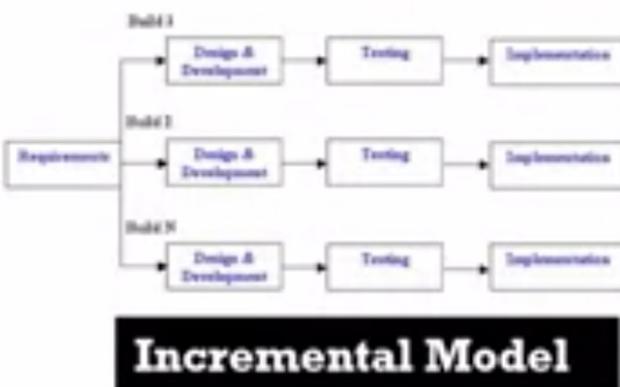
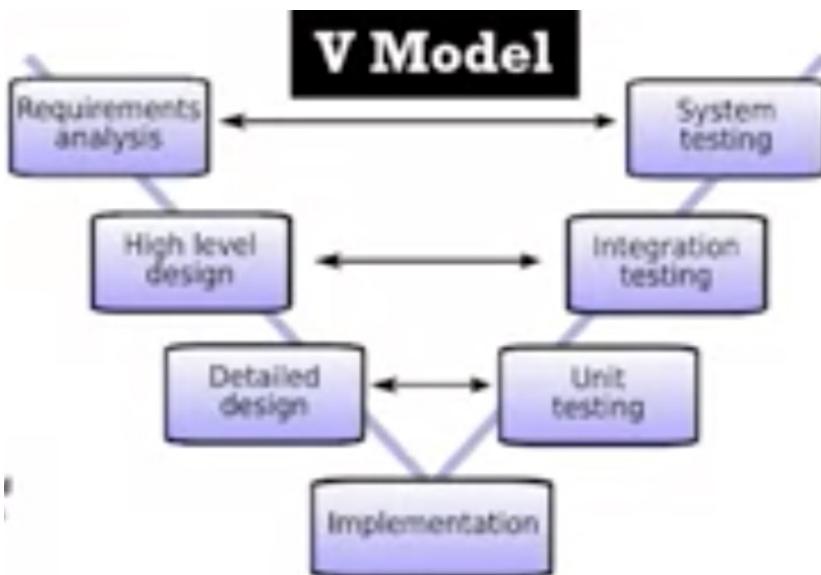


BUILDING SOFTWARE?



WATERFALL MODEL

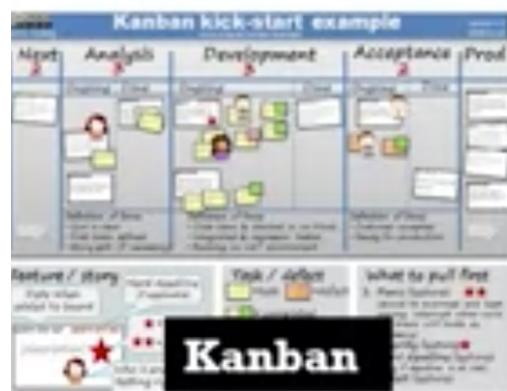




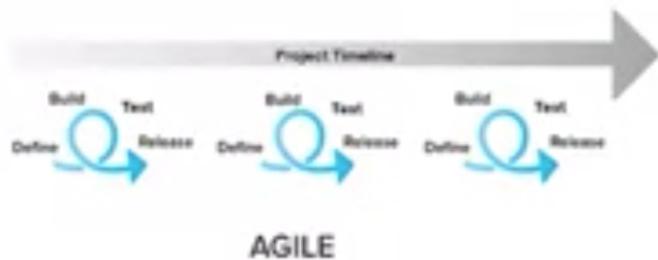
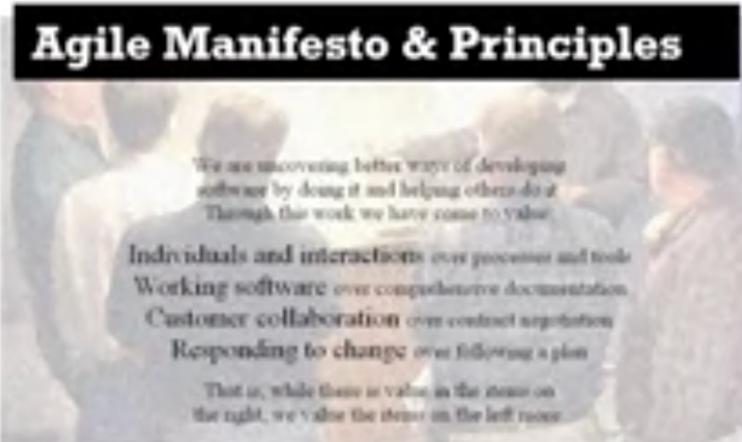
MODELS



Agile



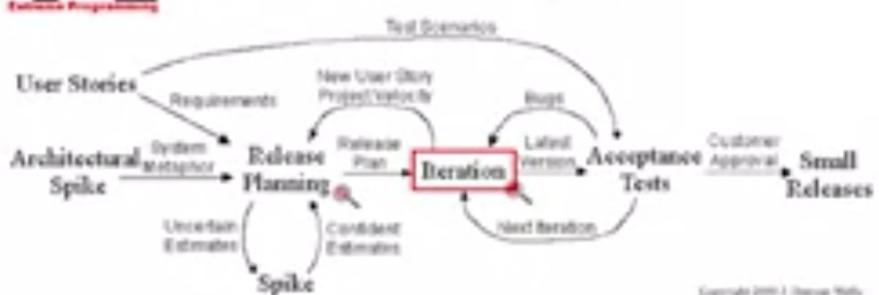
Kanban



AGILE



Extreme Programming Project

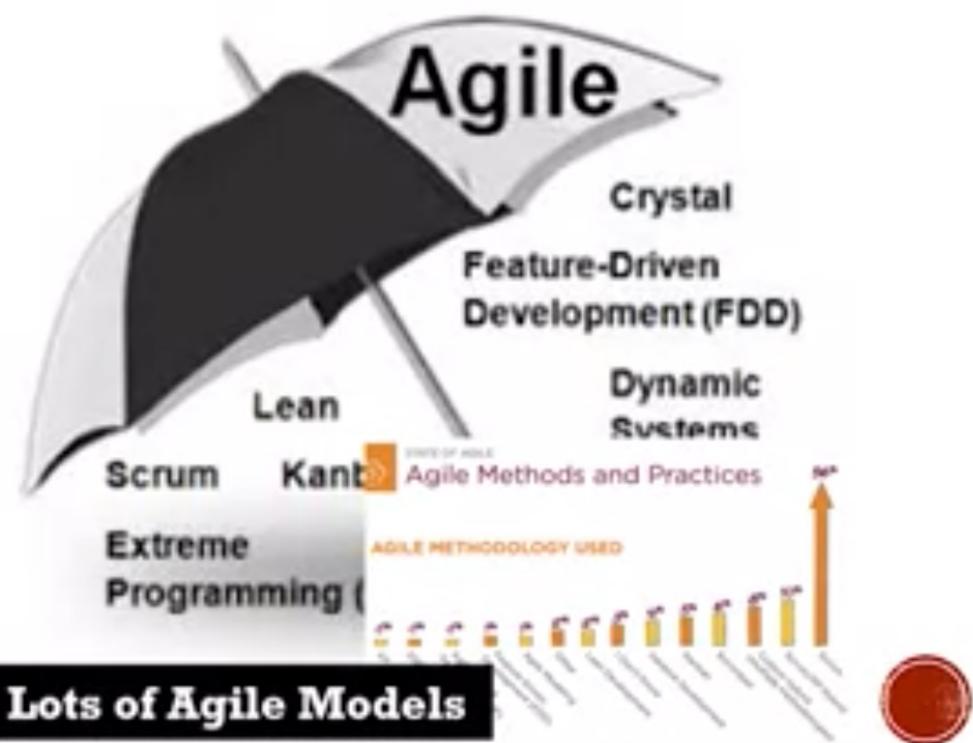


Copyright 2000 J. Cleave Hall

The Agile Framework at a glance



Scrum



Lots of Agile Models

HOW TO RESPOND QUICKLY AND CONFIDENTLY?

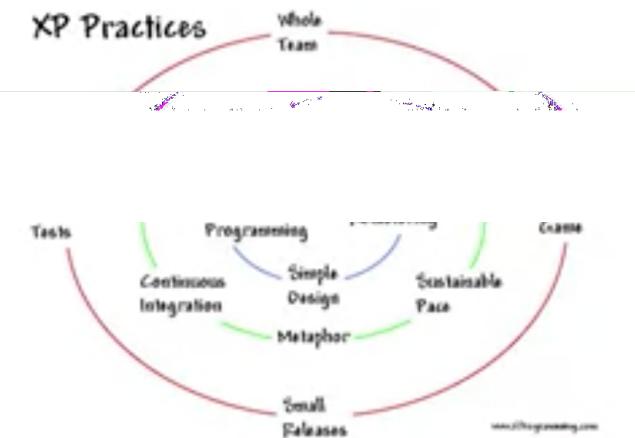
**Continuous
Integration**

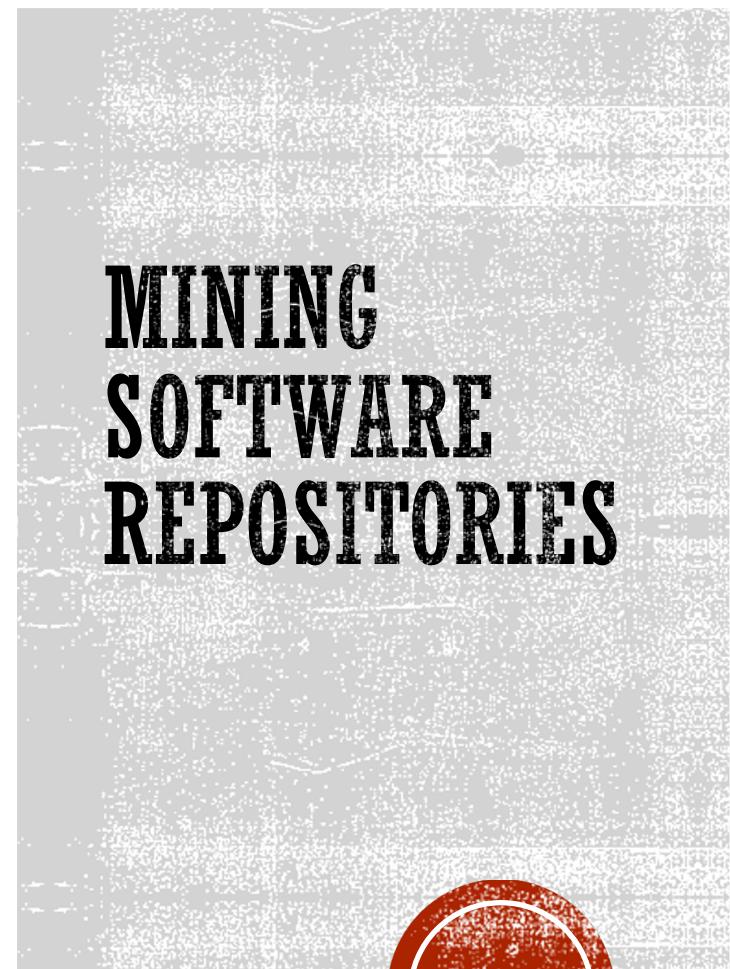
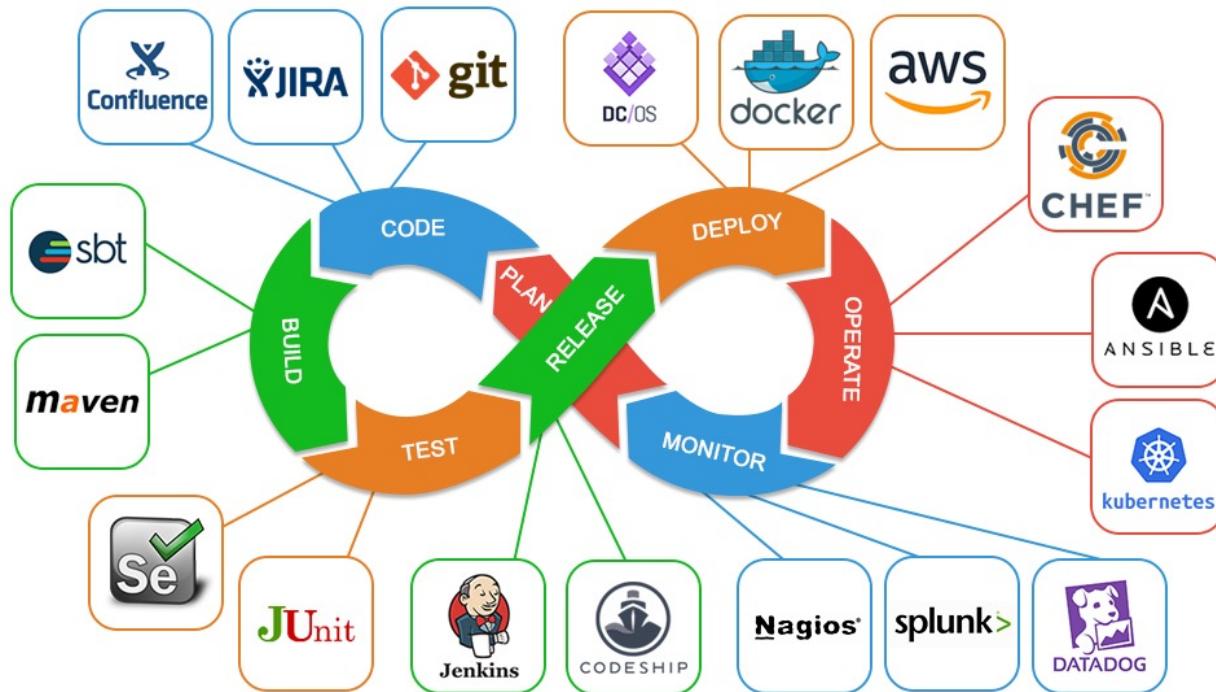
**Automated
Testing**

**Automated
Deployment**

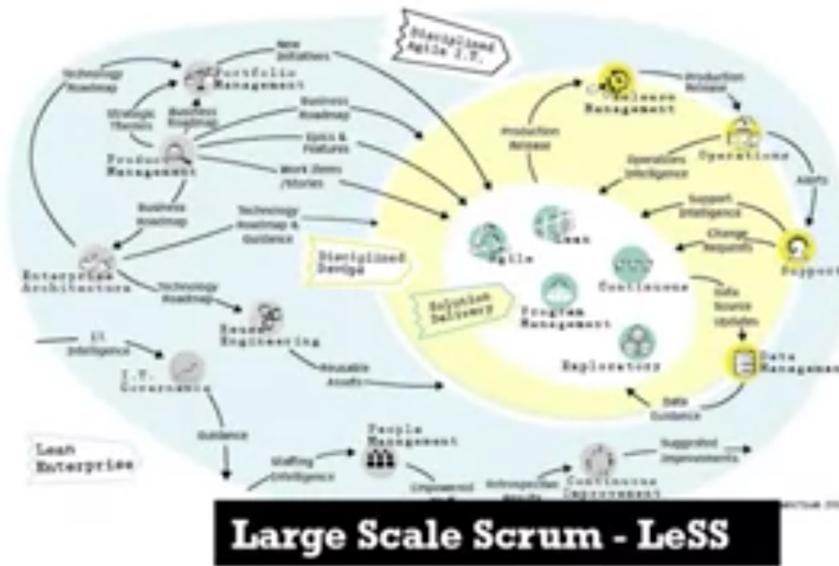
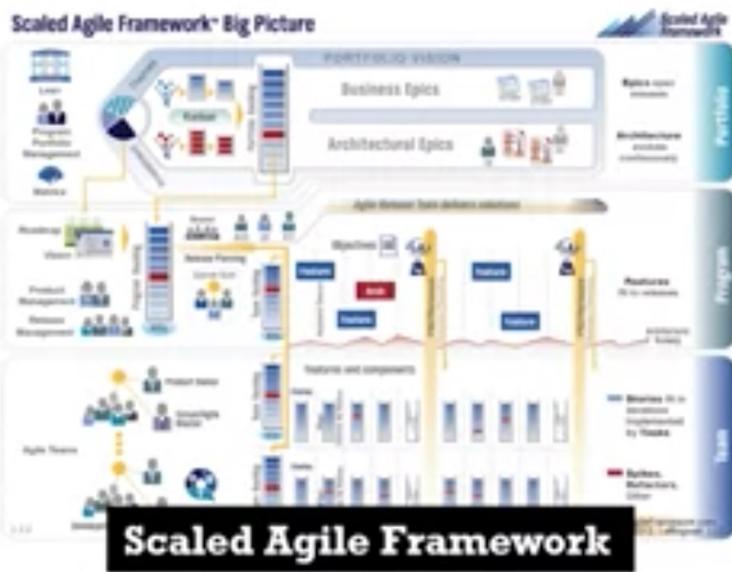
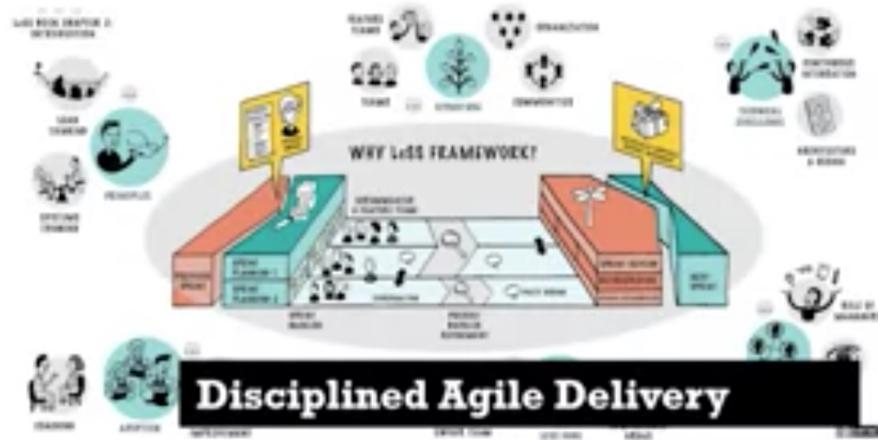


 **Automation**





CAN AGILE WORK FOR BIG PROJECTS?



IS THERE A CHEAPER, FASTER WAY TO LEARN?



Output => Outcome





**PROCESS IS
NOT
SUFFICIENT**



**Let us become a
Software Engineer
(Not same as Coder) !!**

Software engineer is a person who applies the principles of **software engineering** to design, develop, maintain, test, and evaluate computer software.

