ERICSSON

Johan Sjöblom
Product Area Manager
Ericsson WasaLab
Johan.Sjoblom@ericsson.fi

## Contents

- How do requirements behave today?
- CMM - Capability Maturity Model - Req. Mgmt
- The Incremental Development Model



ERICSSON

## How Do Requirements Behave Today?

**Requirements Loosen Up when moving from networks to services and applications**

ERICSSON

## Requirements without 50k of ITU/ETSI specs?

- No Microsoft type player around in Telecom??
- Industry Consortiums:
  - Symbian, EPOC operating system, www.symbian.com
  - WAP Forum, www.wapforum.org
  - Bluetooth
- Other de facto standards
  - IETF: Req. For Comments (RFC's)
  - Java API's
  - W3C (HTML, XHTML, XML)



ERICSSON

## CMM & Requirement Management

- CMM - Capability Maturity Model
- Levels 1-5
- Level 2 is project oriented, and is called "repeatable"
- KPA - Key Process Areas
- Requirement Management is a KPA in CMM level 2

ERICSSON

## CMM 2 - KPA - RM - Activity 1

- The software engineering group reviews the allocated requirements BEFORE they are incorporated into the software project
  - 4. Commitments resulting from the allocated requirements are negotiated with the affected groups.



ERICSSON

## CMM 2 - KPA - RM - Activity 2

- The software engineering group uses the allocated requirements as the basis for software plans, work products, and activities.
  - 1. Allocated requirements are managed and controlled.

ERICSSON

## CMM 2 - KPA - RM - Activity 3

- Changes to the allocated requirements are reviewed and incorporated into the software project
  - 1. The impact to existing commitments is assessed, and <u>changes are negotiated</u> as appropriate.



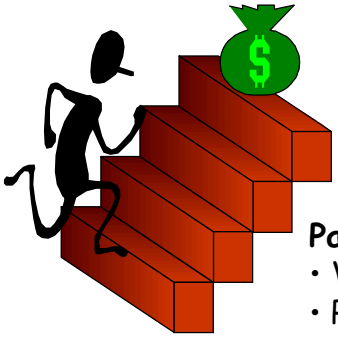
ERICSSON

## Incremental Development

ID

## Introduction to Incremental Development

ERICSSON



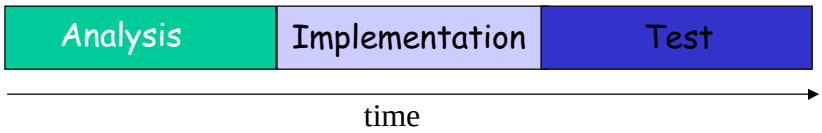
**Part I**
- Waterfall model
- Problems with the waterfall model
- Incremental Development concepts

**Part II**
- Incremental Development:
  Benefits, Pitfalls & Concerns
- Experiences



## Waterfall model

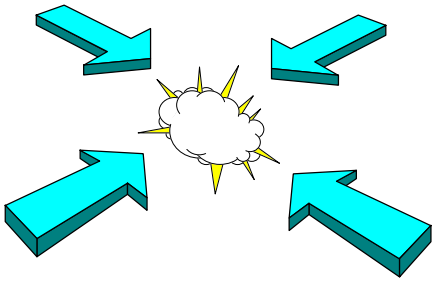
ERICSSON



## Existing Problems (1)

ERICSSON

- Late feedback for customers and designers
- How to cope with changing requirements?
- Big bang integration with interface and integration problems



Big Bang Integration



## Existing problems (2)

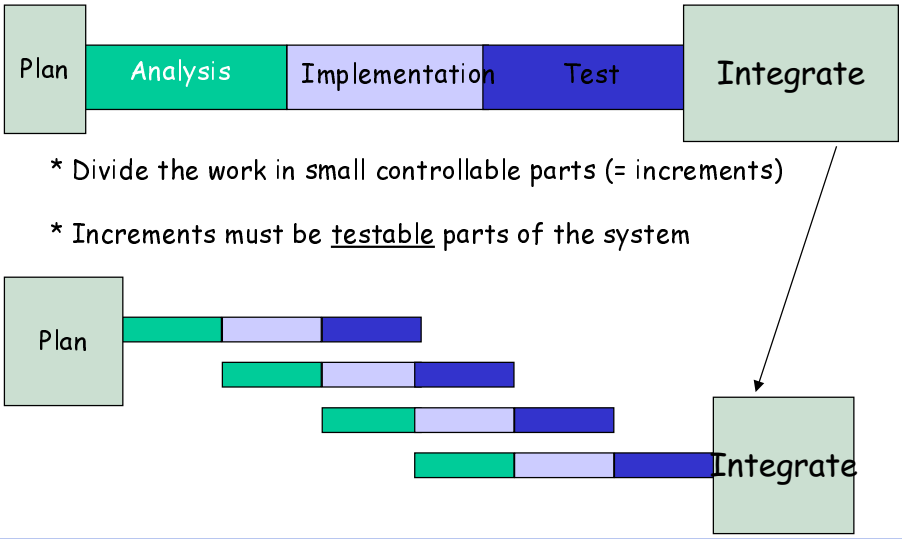
ERICSSON

"Rush on test resources"

- Rush on test resources
- Lack of Project Control
- Slow process improvement

## ID Process View

ERICSSON



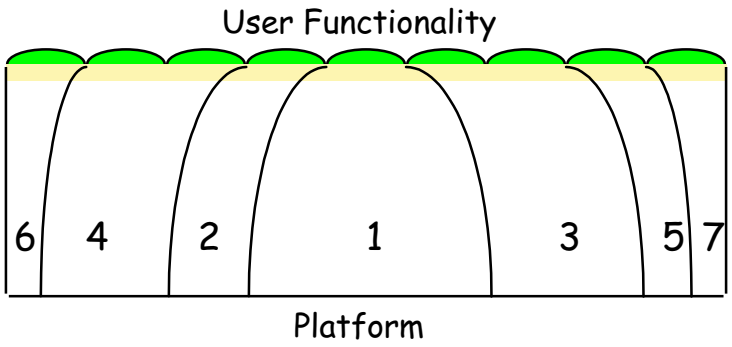

* Divide the work in small controllable parts (= increments)

* Increments must be <u>testable</u> parts of the system



## ID System View

ERICSSON




Each accumulation of developed increments is a complete user executable system

## Benefits & Pitfalls

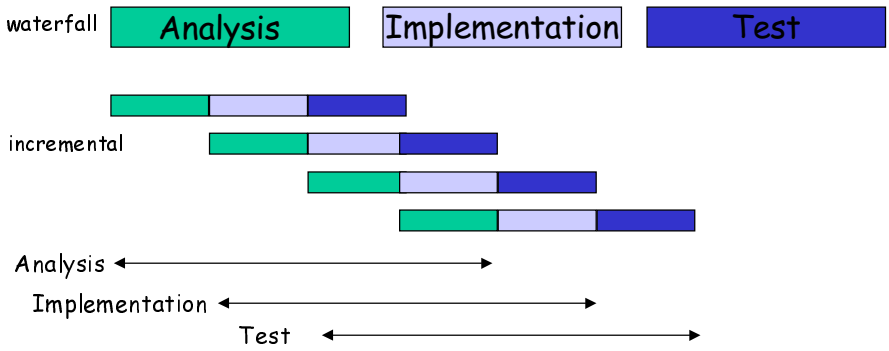
ERICSSON



## Benefits

ERICSSON

1) Lead-time reduction

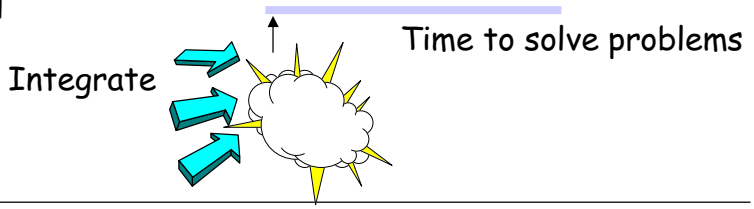


## Benefits

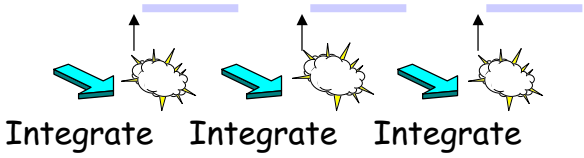
ERICSSON

2) No 'big bang' integration

Waterfall




Incremental Development

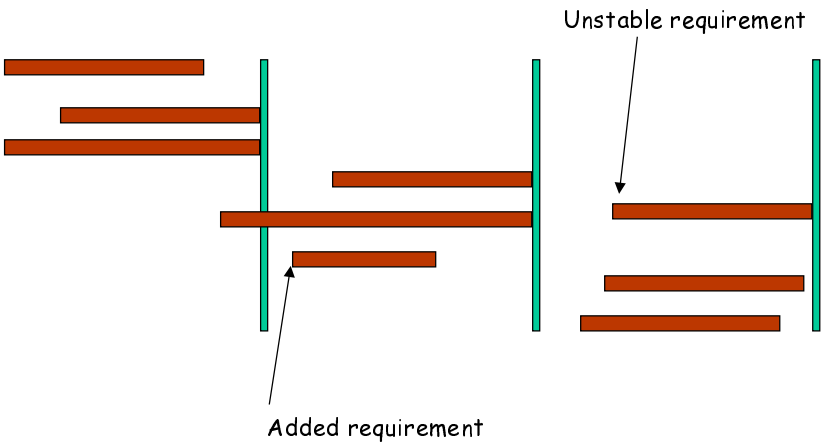





## Benefits

ERICSSON

3) Handling of unstable requirements

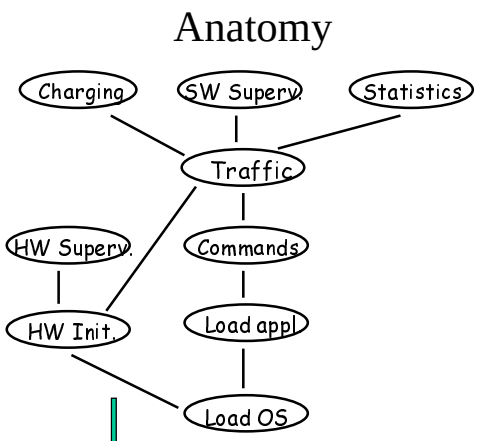


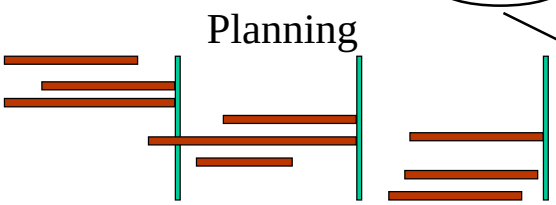
## Pitfalls

ERICSSON

### 1) Planning & Tracking

- Allocation of features dependent of technical contents.
- Delay in one part of the project can on short term impact early deliveries.



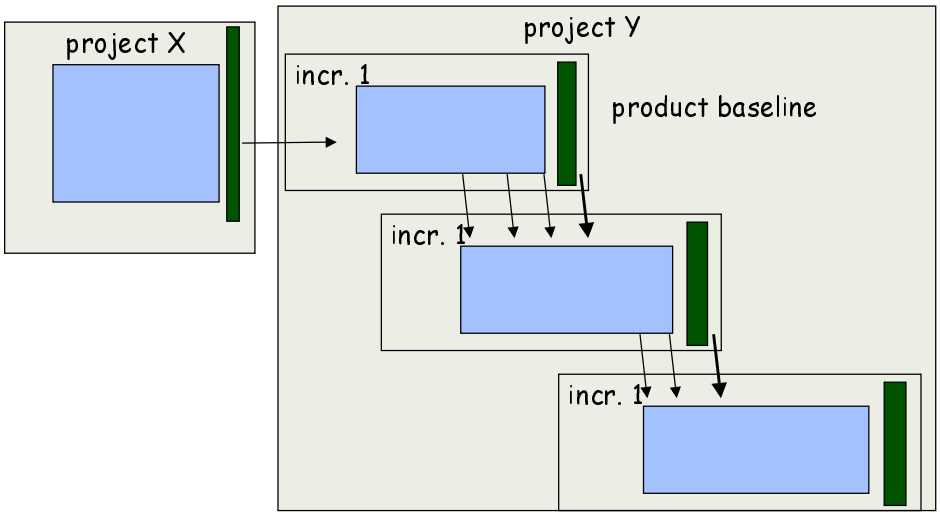






## Pitfalls

ERICSSON

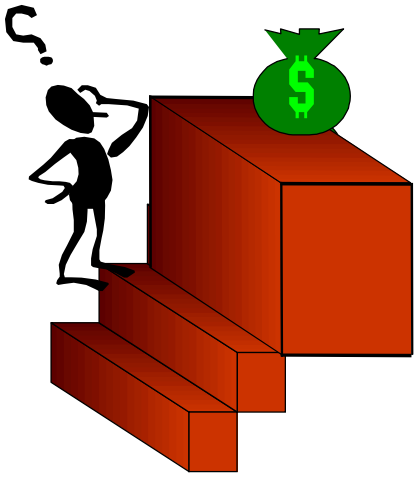
### 2) Configuration Management



## Pitfalls

ERICSSON

3) Postponement of features





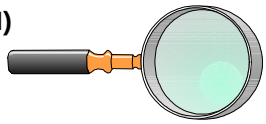
## Concerns

ERICSSON

**More increments implies shorter design - verification cycles**

- **Extra testing**
  - **Test cases needs to be rerun in several increments**
  - **Possibility for continues system test**

- **More reviews and inspections**
  - **Documents impacted by several increments need more inspections. (simplyfied inspections from R&I)**



## Conclusion

ERICSSON

- ID solves the following problems in waterfall projects:
  - Late feedback / Changing requirements / Big bang integration
  - Rush on test resources / Lack of project control / Slow process improvement
- Is has been used successfully for the last decade
- Extra attention needed for:
  - Planning & Tracking
  - Configuration Management
- Increased use of testing and reviews & inspections