Tutorial Software as Integrating Technology in Complex Systems

by Gerrit Muller  Buskerud University College

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

This tutorial describes the integrating value of software in complex systems. The extensive use of software technology to integrate other technologies has a significant impact on the product characteristics and on the product creation organization and process. This tutorial provides insight in the relation between software and the system, and it provides insight in the consequences for the product and the organization. Some recommendations are provided to cope with these consequences.
Program

- Case: the waferstepper and it’s context
- The role of software in general
- Levels of abstraction
- Software -> System Functionality and Qualities
- Requirements perspective
- Evolution and Growth
- Why do we always have problems with software?
- Conclusion
What is a waferstepper

source

reticle

lens

wafer
From stepping to scanning

**stepper:** static exposure of field

**scanner:** dynamic exposure through slit

- stepper:
  - n
  - step
  - n+1

- scanner:
  - reticle
  - slit
  - wafer
  - 250 mm/s

- 250 mm/s

- expose
  - step
  - expose

- t
Key specifications waferstepper

**Imaging**
- Critical dimension: 130 nm
- Line width: 10 nm

**Alignment**
- Overlay: 45 nm
Moore’s law

line width in nm

1000
100
10

1997 1999 2001 2003 2005 2007

250 180 130 100 70 50
Everything influences overlay

Overlay Influence Diagram.
(Maarten Bonnema, 19-3-1999)

- Heat flow from LoS into IF beams and compartment
- Heat flow from SS into RS chuck and compartment
- Metroframe vibration due to water cooling (lens and coolplates)
- Airmount noise, limited vibration isolation
- Metroframe temperature drift
- Effect on showers
- Effect on position of mirrors and IF
- ATHENA Mounting accuracy/stability
- T stability in LS lightpath
- Wafer distortion due to Wafer table/chuck
- Gravity compensation noise
- Heat flow from LoS into IF beams
- Metrology inaccuracy
- Metrology errors
- Servo error
- Feedforward errors
- Wafer expansion by input temperature offset
- Wafer expansion by exposure
- Chuck deformation
- Chuck dimensional stability
- Heat flow from SS into WS chuck
- Chuck expansion
- Chuck dimensional stability
- Heat flow from LoS into IF beams
- Inaccurate lens acceleration
- Feedforward
- P and T of Air, Turbulences
- Data Delay
- ATHENA Measurement accuracy
- Accuracy of lens manipulators
- Lens dynamics
- Reticle Clamping induced distortion
- Reticle heating
- Reticle errors
- Reticle calibration
- Fiducial stability
- Fiducial calibration
- TIS measurement
- Metrology errors
- Servo error
- Feedforward errors
- Heat flow from LoS into IF beams and compartment
- Heat flow from SS into RS chuck and compartment
- Metroframe temperature drift
- Effect on showers
- Effect on position of mirrors and IF
- ATHENA Mounting accuracy/stability
- T stability in LS lightpath
- Wafer distortion due to Wafer table/chuck
- Gravity compensation noise
- Heat flow from LoS into IF beams
- Metrology inaccuracy
- Metrology errors
- Servo error
- Feedforward errors
- Wafer expansion by input temperature offset
- Wafer expansion by exposure
- Chuck deformation
- Chuck dimensional stability
- Heat flow from SS into WS chuck
- Chuck expansion
- Chuck dimensional stability
- Heat flow from LoS into IF beams
- Inaccurate lens acceleration
- Feedforward
Exercise 1, 10 minutes

Make a 3 picture description (What, How, biggest challenge) of your own system.
Fab Context of Waferstepper

- Fab technology
- Waferstepper processing
- Tracks
- Metrology
- Matching
- Contamination
- Fab logistics
- Fab automation
- SPC
- Yield optimization
- Semiconductor design
- Reticule production
- Reticule logistics
- Operator
- Wafer layout
- Fab cost model
- Fab infrastructure
- Utilization
- Inspection and monitoring
- Reticles
- Devices
- Resist
- Fab cost
- Maintenance
Business Context

- Yield
- Value of performance (MHz)
- CD control
- Key driver trade-off

Other players:
- Equipment vendors
- System integrators
- Lease companies
- Fab designers
- Consultants
- Mask makers
- Resist makers
- Wafer makers
- OEM's: Laser
- Intimate partners: Lens

Business models of the customer:
- Design houses
- Foundries
- Vertical integration

Limited number of customers;
Many systems per customer
### Human Context: Stakeholders

#### "external"
- customer
- purchaser
- decision maker
- user
- operator
- maintainer

#### "internal"
- managers
  - business manager
  - marketing manager
  - product manager
  - operational manager
  - project leader
  - sales manager
  - quality manager
  - logistics manager
  - line manager
  - technology manager

- engineers
  - system engineers
  - experts
  - manufacturing engineers
  - customer support

#### other
- government
- customer's customer
- banks, insurance

#### suppliers
- component manufacturer
- outsourced design
Multitude of Disciplines
Tutorial Software as Integrating Technology in Complex Systems

Gerrit Muller

version: 0.1
March 6, 2013
INSEcomplexity
Symptom: Delays appear during Integration

Do you have any design issues for the design meeting?

The default answer is: No.

During integration numerous problems become visible.
Exercise 2, 10 minutes

Make a 3 picture description (Application context, Value chain, technologies) of your own system.
Relative Contribution of SW

- SW
- electronics
- mechanics
- physics / chemical etcetera

1970

2000

Relative effort

100%
Characterization of disciplines

- Concrete
- Tangible
- Mature
- Production lead-time
- Material cost

- Abstract
- Intangible
- Immature
- Flexible?

Mechanics
Analogue / power Electronics
Digital Electronics
Software
Exponential Pyramid, from requirement to bolts and nuts

- \(10^0\): number of details
- \(10^1\): system requirements
- \(10^2\): design decisions
- \(10^3\): parts
- \(10^4\): connections
- \(10^5\): lines of code
- \(10^6\): multi-disciplinary
- \(10^7\): mono-disciplinary

Research focus:
- System
From Components to System Qualities

- **components**
  - source
  - mirrors
  - fiducials
  - lens elements
  - flaps
  - air showers

- **subsystems**
  - frames
  - motors
  - sensors
  - robot
  - bolts
  - nuts
  - air mounts
  - PCBs
  - ICs
  - cables
  - cabinets

- **functions**
  - prepare
  - align and calibrate
  - scan and expose
  - transport wafer

- **system qualities**
  - overlay
  - CD control
  - productivity

- **system qualities layers**
  - transport reticle
  - electronics infra
  - metro frame

- **levels**
  - prepare
  - scan and expose
  - transport wafer
  - align and calibrate
  - prepare

- **system**
  - prepare
  - scan and expose
  - transport wafer
  - align and calibrate
  - prepare

- **levels**
  - prepare
  - scan and expose
  - transport wafer
  - align and calibrate
  - prepare

Tutorial Software as Integrating Technology in Complex Systems

version: 0.1
March 6, 2013

Embedded Systems

Gerrit Muller
Role of Software

- **components**
- **subsystems**
- **functions**
- **system qualities**

**SW** implements functionality and determines emerging qualities.
Exercise 3, 10 minutes

Make a toplevel decomposition of the software in your system and estimate the amount of software of the constituting parts
When SW engineers demand "requirements",
then they expect *frozen* inputs
to be used for
the design, implementation and validation
of the software
System vs Software Requirements

The diagram illustrates the relationship between system requirements and software requirements. It shows a pyramid with the y-axis labeled as 'number of details' and the x-axis showing different levels of complexity.

- **System Requirements**
  - Multi-disciplinary
  - Mono-disciplinary

- **Software Requirements**
  - Mono-disciplinary

The pyramid starts from the bottom, which represents the highest level of detail (10^7), and moves upward as the level of detail decreases (10^0). This visual representation helps in understanding the scope and complexity of requirements in complex systems.
Why is the Software Requirement Specification so Large?

- software subsystem
- user interface
- system behavior
- operational choices
  - synergy, tools, ...
- limited computing resources
- control of physical subsystems:
  - sensors, actuators

Diagram:
- Software subsystem
- User interface
- System behavior
- Operational choices
  - Synergy, tools, ...
- Limited computing resources
- Control of physical subsystems:
  - Sensors, actuators
And why is it never up-to-date?
Exercise 4, 2 minutes

How many pages are in your Software Requirements Specification?
Control Hierarchy of a Waferstepper
Frequency of Control Actions

trend with increasing performance requirements

- SW sampling per wafer per die per batch per day preventive maintenance
- 10^{-3} 1 10^3 10^6

seconds
Evolution of System Control

1990
150 kloc

2000
2000 kloc
Consequences of Evolution

- **Performance and functionality demands**
- **Complexity**
- **Reliability**

- Loss of overview (150kloc fits in 1 mind, 2Mloc not)
- Exponential increase of coupling
- 1:1 relation HW:SW becomes n:m relation

- Autonomous subsystems
- Paradigm shift!
- Integrated system

---

Tutorial Software as Integrating Technology in Complex Systems

version: 0.1
March 6, 2013
FAIevolutionConsequences
Exercise 5, 10 minutes

Visualize the (SW) evolution of your system. What is your current phase?
Different Focus of Software and System

**System engineering focus**
- **qualities**
  - productivity
  - image quality
  - reliability
- **concerns**
  - integral design (quality, balance)
  - system context
  - lifecycle
  - operational processes
- **education**
  - principles
  - heuristics
  - analysis and synthesis
  - processes

**SW engineering focus**
- **qualities**
  - functionality
  - maintainability
  - variability
- **concepts**
  - structure
  - (generic) mechanisms
- **concerns**
  - configuration management
  - release procedure
  - tools
  - SW processes
  - SW problems, change requests
- **education**
  - principles
  - heuristics
  - analysis and synthesis
  - processes
  - languages
  - operating systems
  - algorithms
  - formal methods
Caricature of a SW Architecture

Property editor

NameSpace server

Event manager

Registry

Persistent Storage

Session manager

Broker

Transparant Communication

Monitor

Compliance profile

Device independent format

Spool server

Queue manager

Resource scheduler

Configurable pipeline

Application

Abstraction Layer

Plug & play

Compliance profile

Compliance profile

Application
Caricature of Physics Systems View

- Laser
- Illuminator
- Sensor
- Pulse freq, bw, wavelength, ...
- Uniformity
- Reticle
- NA
- Abberations
- Transmission
- Lens
- Aerial image
- Wafer
Relation SW and Physics

- laser
- illuminator
- sensor
- reticule
- aerial image
- NA
- abberations
- transmission
- lens
- wafer

Measure
Adjust
Calibrate
Analyse
Process
Log
Control

- pulse-freq, bw, wavelength, ..
- uniformity

Tutorial Software as Integrating Technology in Complex Systems

version: 0.1
March 6, 2013
TSAITphysicsAndSW
**Symptoms of too isolated SW efforts**

*symptoms*

SW people are clustered together

SW is alpha tested before system integration

SW team uses own specification and design process

SW specification is in SW jargon or formalism

*counter measures*

colocation per function, subsystem or quality

continuous system integration

higher level processes are shared

interaction between SW, HW and system engineers
What is the degree of integration or isolation of SW in your organization?
Different Mindsets and Characteristics

**System**

product: sellable self-sustained entity operating in a broader context

**different focus:**

"qualities"
"concerns"
"concepts"
"education"

**HW engineering**

tangible
concrete
goods flow costs & lead times
physics laws

**SW engineering**

intangible
abstract
no goods flow costs
"everything is possible"