Component-based Software Engineering
Process and Component Lifecycle

Ivica Crnkovic
Mälardalen University,
Department of Computer Engineering
Sweden
ivica.crnkovic@mdh.se
http://www.idt.mdh.se/~icc
Software Development processes

• What determines which development process model to use?
  – Type of products/products (requirements from customers)
  – Type of project
  – Availability & requirements of stakeholders
  – Type of organization
  – Technology used
  – …..
Software development process adaptation

- Software development processes are usually of generic type
  - Usually requires adaptations
  - Often a software development process is a combination of several models
  - There is difference between theory and practice
    - Practice is often more complex
    - Practice is not perfect
Lifecycle Process Models for products

Concept → Development → Production → Utilization → Retirement

Generic Product Lifecycle
Lifecycle Process Models for software products
Component-based approach process characteristics

- Separation of the development process.
  - The development processes of component-based systems
  - Development processes of the components.

- A new process: Component Assessment.
  - Finding and evaluating the components.

- Changes in the activities in the development processes.
  - System-level process the emphasis will be on finding the proper components and verifying them,
  - Component-level process, design for reuse will be the main concern.
A simplified and an idealized process

• Assumption of the model
  – components selected and used are sufficiently close to the units identified in the design process

• Further, the figure shows only the process related to the system development – not to the supporting processes
CB System Development…

- Requirements
- Design
- Implementation
- Integration
- Test
- Release
- Maintenance

System Development
CB system and Components Development

- Requirements
- Design
- Implementation
- Integration
- Test
- Release
- Maintenance

Component Development

System Development
The complete process

Requirements
Design
Test
Release
Maintenance
Implementation
Integration
Find
Select
Verify
Store
Test
Release
Maintenance
System Development
Component Assessment
Component Development

Requirements
Design
implementation
Integration
Test
Release
Maintenance

MÄLARDALENS HÖGSKOLA
MRTC
MÄLARDALENS REAL-TIME RESEARCH CENTRE
System Requirements Phase

- Collect requirements
- Analyse requirements
- Specify/ refine requirements
- Are there components that fulfill requirements?

Component A
System and Analysis & Design Phase

System analysis

Conceptual design

Detailed design

Architectural components

Existing components
Different architecture view in different phases

• Phase I
  – System architecture - Decomposition of the system
System Design – Phase 2

- Implementation Architecture - Component Identification
System Design – Phase 3

• Deployment architecture

![Deployment Architecture Diagram]
System Implementation Phase

- Implementation
  - Selection of existing components
  - Implementation of new components
  - Glue-coding

- Adaptation

- Parameterized Interface
- Wrappers
- Adapters
• **Parameterized Interface.** Parameterized interface makes it possible to change the component properties by specifying parameters that are the parts of the component interface.

• **Wrapper.** A wrapper is a special type of a glue-code that encapsulates a component and provides a new interface that either restrict or extend the original interface, or to add or ensure particular properties.

• **Adapter.** An adapter is a glue code that modifies (‘adapts’) the component interface to make it compatible with the interface of another component.
System Integration Process

• Putting components together
• Integration components into the system (component) framework
• Integration can happen in different phase of product’s lifecycle
Integration in different phases

- **Modeling/design**
- **Assembly time (link)**
- **Run-time**

**Assembly – architectural components**

Integration of components (for testing feasibility)

System building from all components

Dynamic updating of components
Test Phase

• System is being verified against the system specification
• In the waterfall model the test is performed after the system integrations,
• In CBD
  – Tests performed for isolated components
  – Tests of assemblies
  – Test of the system
• Tests are present in all phases!
Integration and test in different phases of the CBD process

- Requirements
  - Design
    - Selection of the component candidates
      - Components Integration
        - Components and Assemblies test
  - Selection
    - Glue-coding
    - Adaptation
    - Implementation of new components
      - Components Integration
        - Components and Assemblies test
      - Integration
        - Test
        - Release
          - Maintenance
            - Selection
            - Adaptation
            - Component updates
            - Components maintenance

Integration and test in different phases of the CBD process.
Release Phase

• packaging of the software in forms suitable for delivery and installation.
• The CBD release phase is not significantly different from a “classical” integration.
System Maintenance Phase

• The approach of CBD is to provide maintenance by replacing old components by new components or by adding new components into the systems.

• The paradigm of the maintenance process is similar to this for the development:
  – Find a proper component, test it, adopt it if necessary, and integrate it into the system.
Component assessment process

Requirements → Design → Implementation → Integration → Test → Release → Maintenance

System Development

Component Assessment

Component Development

- Find
- Select
- Verify
- Store

Requirements
Design
Implementation
Integration
Test
Release
Maintenance

Component Development

MÄLARDALENS HÖGSKOLA
MÄLARDALEN REAL-TIME RESEARCH CENTRE
A generic assessment process activities

• **Find** – From an “infinite” component space find the components that might provide the required functionality.

• **Select** – Between the components candidates found, select a component that is most suitable for given requirements and constraints.

• **Verify** –
  - Test functional and certain extra-functional properties of a component in isolation.
  - Test the component in combination with other components integrated in an assembly.

• **Store** –
  - store the selected components in a component repository.
  - Store additional specification (metadata)
    - measured results of component performance,
    - known problems,
    - the tests and tests results and similar
A assessment process activities

• Activities
  – Find
  – Select
  – Verify
  – Store

• The concrete activities dependent of type of component-based development process
  – Architecture-driven component development
  – Product-line development
  – COTS-based development.
Component development process

- Requirements
- Design
- Test
- Release
- Maintenance
- Implementation
- Integration

System Development

Component Assessment

Component Development

MRTI

MÄLARDALENS HÖGSKOLA
Component development process - specifics

• Components are built as reusable units
• Components are built for future systems
• Consequences
  – There is greater difficulty in managing requirements;
  – Greater efforts are needed to develop reusable units;
  – Greater efforts are needed for providing component specifications and additional material that help developers/consumers of the components.
Different architectural approaches in CBD

- Architecture-driven component development
- Product-line development
- COTS-based development
- Subcontractor-based development
Architect-driven component development

• Top-down approach
  – components are identified as architectural elements and as a means to achieve a good design.
  – Components are not primarily developed for reuse,
  – Component-based technologies are used, because of easier implementations, in getting already existing serviced provide from the component technology.
  – the main characteristic of these components is composability,
  – No emphasis on while reusability
  – No emphasis time-to-market issues
Architect-driven component development

– The parallel development processes are reduced to two semi-parallel processes
A product line is:

* From Rob Van Ommering/Philips
Product-line development

System Development

Component Development

Requirements

Design

Implementation

Integration

Test

Release

Verify

Store

Maintenance

Select

Component Assessment
COTS-based development

- COTS - commercial off the shelf
- component development process completely separately developed from system development.
- The strongest concern
  - time-to-market from the component user point of view,
  - reusability from the component developer point of view.
- Main characteristics
  - instant value of new functionality
  - Challenges in composability
    - Often COTS components don’t comply with a component model,
    - the semantics of the components are not specified
    - different properties of the components are not properly and adequately documented.
Subcontractor-based development

Every system solution must be optimized

How to reuse components and achieve Optimized solutions?

Fig. 3. Schematic block diagram of the HIL electronics.
Subcontractor-based development

System Development

Component Assessment

Component Development

Requirements

Design

Implementation

Integration

Test

Release

Maintenance

Select

Verify

Adjust

MRTCS

MÅLARDALENS HÖGSKOLA
Case Study

• Philips Consumer Electronics (TV, Video, DVD)
  – Moving from a hardware local development to software & hardware global development

• Requirements
  – New products (product models, variants, etc.) must be delivered each 6 months

Experience
  “hardware-like” componentization of software made it possible to make the transformation
The domain…

* From Rob Van Ommering/Philips
(1) Complexity

Code Size Evolution of High End TV Software

* From Rob Van Ommering/Philips
(2) Diversity

- Connectivity
  - 1394
  - P50

- Broadcasting Standard
  - DTV
  - TiVo
  - TVCR
  - HD
  - DVD

- Storage Device
  - VCR
  - DVD

- Video Output Device
  - PTV
  - FTV
  - LCTV

- User Interface
  - EPG
  - Menu
  - Animation

- Data Processing
  - Price
  - Image

- Sound
  - 100 Hz
  - Dolby
  - AC3

- Region
  - Eu
  - US
  - AP

- *From Rob Van Ommering/Philips*
(3) Lead Time

Was:
• Yearly cycle of product introduction
  – Christmas
  – World championship

Is:
• Decreasing to 6 (or even 3) months
  – Otherwise loose shelf space in shop
Product architecture:

- Operating system
- Platform – component framework
- Core components
- Application components
Koala Components

* From Rob Van Ommering/Philips
Development development process

- Overall architecture development
- Platforms
- Subsystems development
- Components
- Product development
CB development requires changes in the organizations!
Experience from Philips

- The new (CBD) approach did not work with the previous development process model and organization
- A lot of efforts has been put on
  - re-organization
  - Emphasis on early system/components specification
  - Quality assurance
    - Early test and verification of the components
  - Synchronization of the activities
Findings from the case study

– CBD requires specific approach in development process
– Reusue is not only a matter of a good technology but also of the process and organisation
– Many companies introducing CBD are not aware of that
Conclusion

• The main characteristic of component-base development process
  – a separation (and parallelization) of system development from component development.

• Consequence
  – Programming issues (low-level design, coding) are less emphasized
  – verification processes and infrastructural management requires significantly more efforts.
Literature

- Ivica Crnkovic, *Component-based Development Process and Component Lifecycle* (a chapter in a never completed CBSE book)