# G Frameworks

## G.1 Overview

*Design is hard.*
*One way to avoid the act of design is to reuse existing designs*
*[Ralph Johnson]*

- Class libraries
- Frameworks — What they are, How they work, Benefits
- Types of Frameworks
- CORBA & Frameworks
- Java Frameworks




## G.2 References

JoF88. R.E Johnson, B. Foote: Designing Reusable Classes, *Journal of Object-Oriented Programming*, June 1988, ftp://st.cs.uiuc.edu/pub/papers/frameworks/designing-reusable-classes.ps.

Joh91. R.E Johnson, Reusing Object-Oriented Design, University of Illinois, Technical Report UIUCDCS 91-1696, 1991, ftp://st.cs.uiuc.edu/pub/papers/frameworks/reusable-oo-design.ps.

Joh92. R.E Johnson, Documenting Frameworks Using Patterns, OOPSLA '92 Proceedings, 1992, ftp://st.cs.uiuc.edu/pub/patterns/papers/documenting-frameworks.ps

Joh93. R.E Johnson, How to Design Frameworks, Tutorial Notes, OOPSLA '93, Washington, 1993, ftp://st.cs.uiuc.edu/pub/papers/frameworks/OOPSLA93-frmwk-tut.ps

OHE96. Robert Orfali, Dan Harkey, Jeri Edwards: *The Essential Distributed Objects Survival Guide*, John Wiley, New York, 1996.

SFO97. San Francisco Project Technical Summary, http://www.javasoft.com/javareel/isv/IBM/SanFrancisco/prod_summary.html

Tal93. Taligent Inc., Leveraging Object-Oriented Frameworks, A Taligent White Paper, 1993

Tal94. Taligent Inc.,Building Object-Oriented Frameworks, A Taligent White Paper, 1994, http://www.taligent.com/Technology/WhitePapers/BuildingFwks/BuildingFrameworks.html

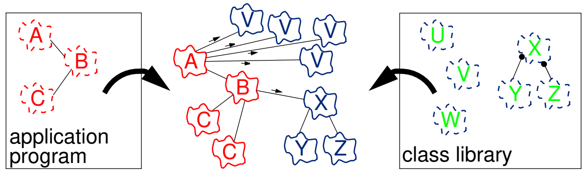
Tal95. Taligent Inc, The Power of Frameworks - For windows and OS/2 developers, Addison-Wesley, 1995




# G.3 Frameworks — What, How & Why

## 1 Classes and Class Libraries

- Class = design for a set of objects
- Class library
  - Collection of classes
  - Flow of control: application objects → library objects







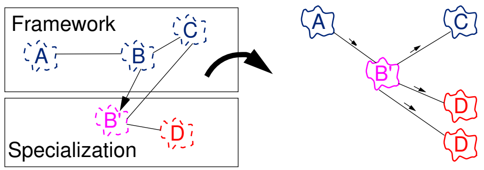


## 2 Frameworks (What)

- Framework = design for a set of applications
  - ➥ design of a set of objects that collaborate to carry out a set of responsibilities
  - ➥ a way to reuse high-level design
- Framework =
  - set of classes
  - \+ rules how the objects play together
  - \+ definition of the interfaces in the game (how can I join)
  - \+ definition of interfaces to the game (interaction with the outside world)
  - \+ definition of the goals of the game
- Compared with a hardware board
  - the board = instance of the Framework
  - ICs = objects
  - backplane = ORB

## 2 Frameworks (What - 2)

- Frameworks
  - ◆ are an application or application skeleton
  - ◆ application developer may
    – add – substitute – modify
    components
  - ◆ Flow of control:
    framework → application object → framework
    **"Don't call us, we'll call you"** (Hollywood principle)







G.3 Frameworks — What, How & Why

## 3 Frameworks (How)

- Two sorts of interfaces, two ways for customization

▲ Client API
- ➤ external interface of the framework
  (how can other applications interact with the framework)
- ➤ described in IDL

▲ Framework API
(Black-box Interface for customization)
- ➤ internal interface of the framework
  (how can new components interact with the rest of the framework
  + how does the framework interact with the new components)
- ➤ interface described in IDL, protocol for registration & notification

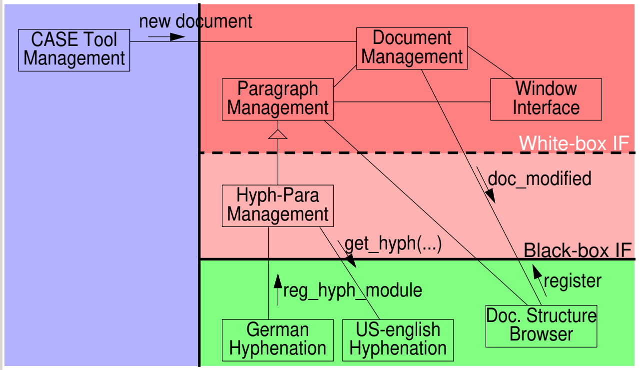
▲ Subclasses of framework components
(White-box Interface for customization)
- ➤ customization + replacement of components of the framework
- ➤ polymorphism guarantees interoperation




G.3 Frameworks — What, How & Why

## 4 Example

- Framework for document processing







G.3 Frameworks — What, How & Why

## 5 Benefits

- Prefabricated infrastructure
  - ➤ reduces coding, debugging & testing
- Architectural guidance
  - ➤ software is wired and ready to go
  - ➤ you just have to plug in your extensions
- Less monolithic applications
  - ➤ small pieces of applications are plugged into existing frameworks
  - ➤ existing frameworks are plugged together
- Foundation for a software components industry
  - ➤ Well-designed general frameworks are the basis for problem-specific solutions
- Reduced maintenance
  - ➤ Frameworks provide the bulk of (hopefully well-tested) code

## G.4 Types of Frameworks

### 1 Application Frameworks

- Expertise applicable to a wide variety of programs
  - ➤ graphical user interfaces

### 2 Domain Frameworks

- Expertise in a particular problem domain
  - ➤ manufacturing control
  - ➤ document processing

### 3 Support Frameworks

- System-level services
  - ➤ file systems, device interaction, ...





## G.5 CORBA & Frameworks

- Main goal of CORBA
  - ➥ infrastructure for business objects

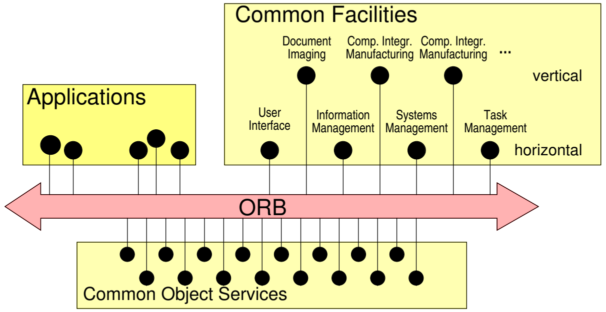
★ Business objects
  - ➤ a representation of a thing active in the business domain
  - ➤ includes
    - – business name and definition
    - – attributes, behavior, relationship, constraints
  - ➤ examples:
    - – a person (customer), a place, a concept (invoice, contract), …
  - ➥ may be used in unpredictable combinations
  - ➥ is independent of specific applications
  - ◆ represents a "everyday life entity" → exists in the "end user's world"
  - ◆ in contrast: entities that make sense only to information systems





## G.5 CORBA & Frameworks (2)

- CORBA Facilities
  - ➤ provide services for business objects
  - ➤ are built on top of Common Object Services
  - ➤ extend the "primitive" COS








## G.6 Java Frameworks

- Java Media Framework
  - ➤ audio and video device control
- Lightweight UI Framework
  - ➤ Customizable user interface environment
- General Administrative Framework

★ Fundamental concepts for building Frameworks with Java:
  - ➥ Interfaces to describe type conformance
  - ➥ Component technology: Java Beans