Thin Client for Web Using Swing

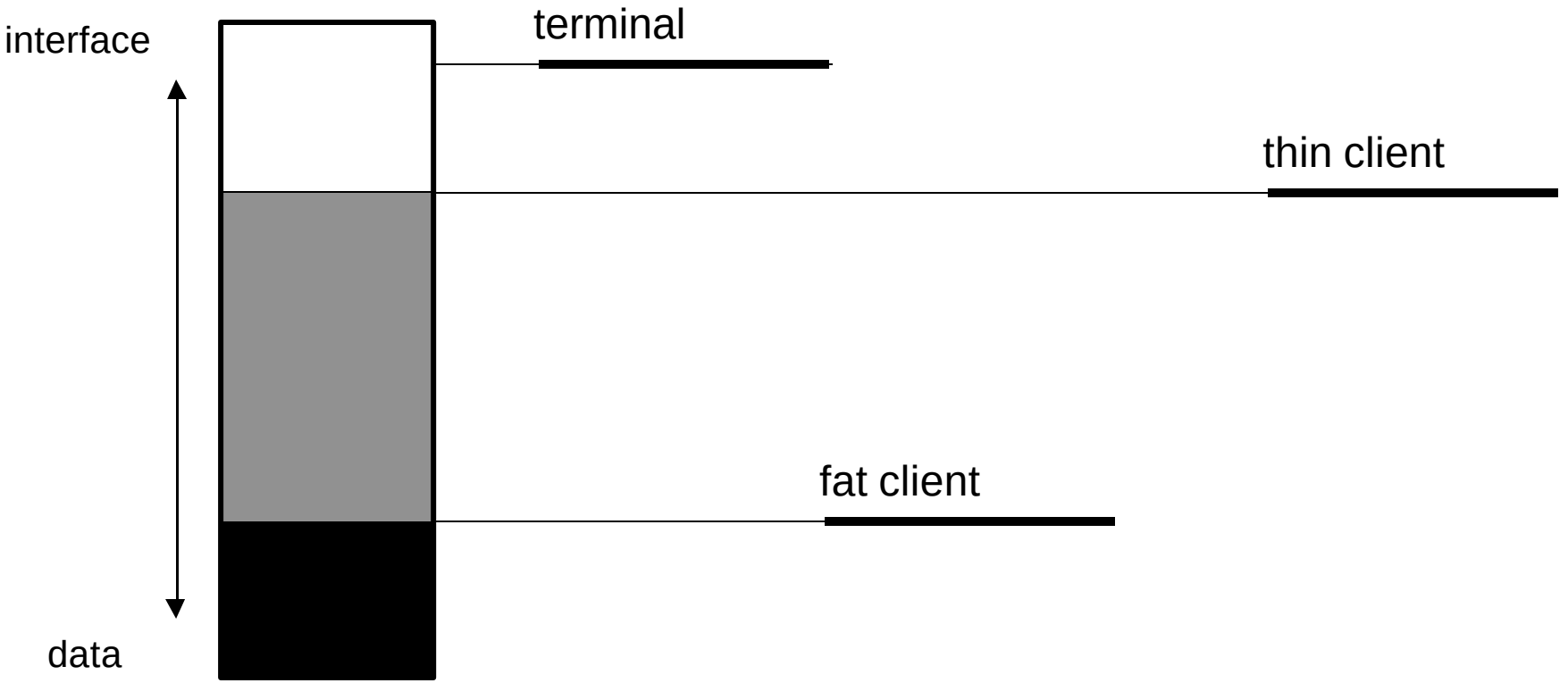
Raffaello Giulietti & Sandro Pedrazzini

SUPSI, Dip. di Informatica ed Elettrotecnica
Galleria 2, 6928 Manno, Switzerland

TINET SA, 6928 Manno, Switzerland

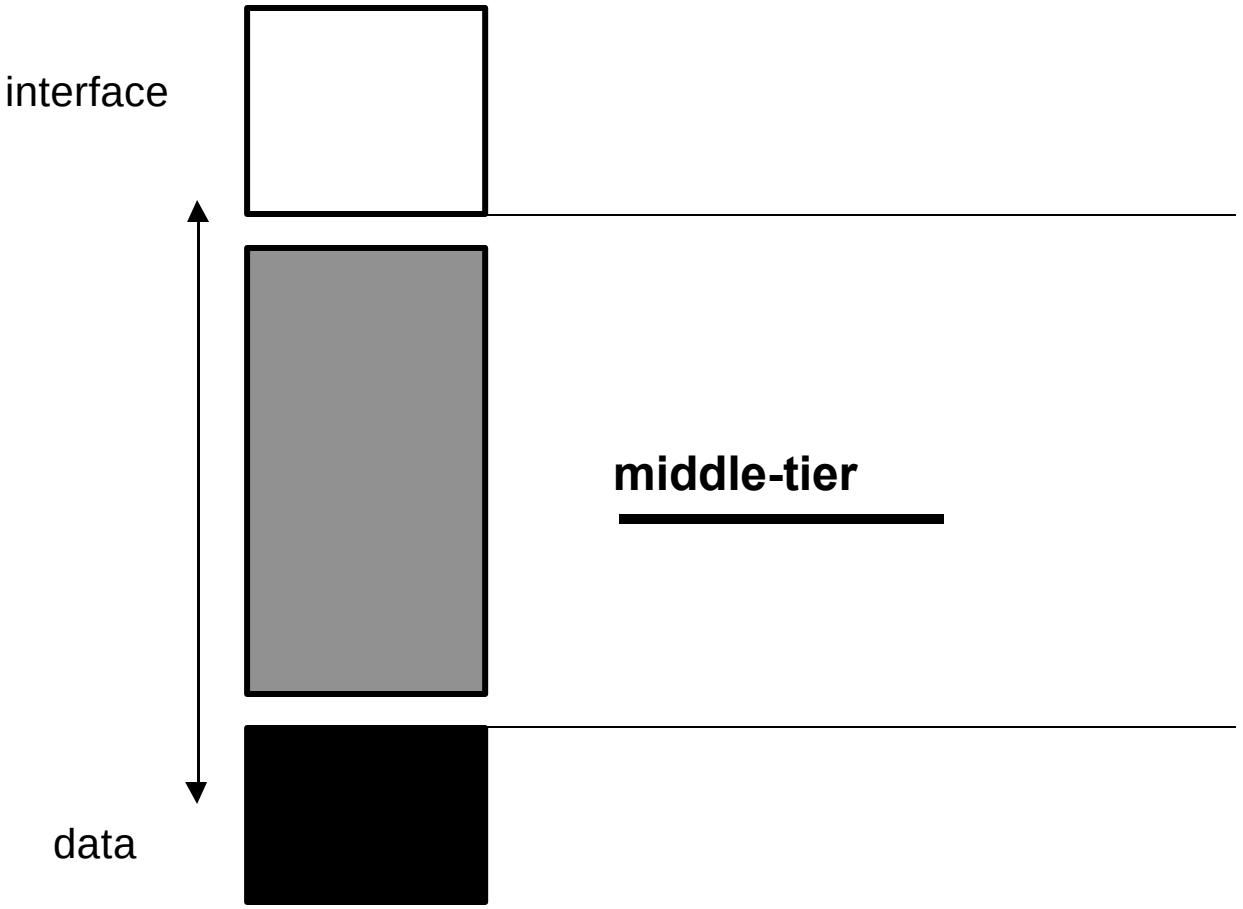
{lello, sandro}@idsia.ch

Evolution client/server model



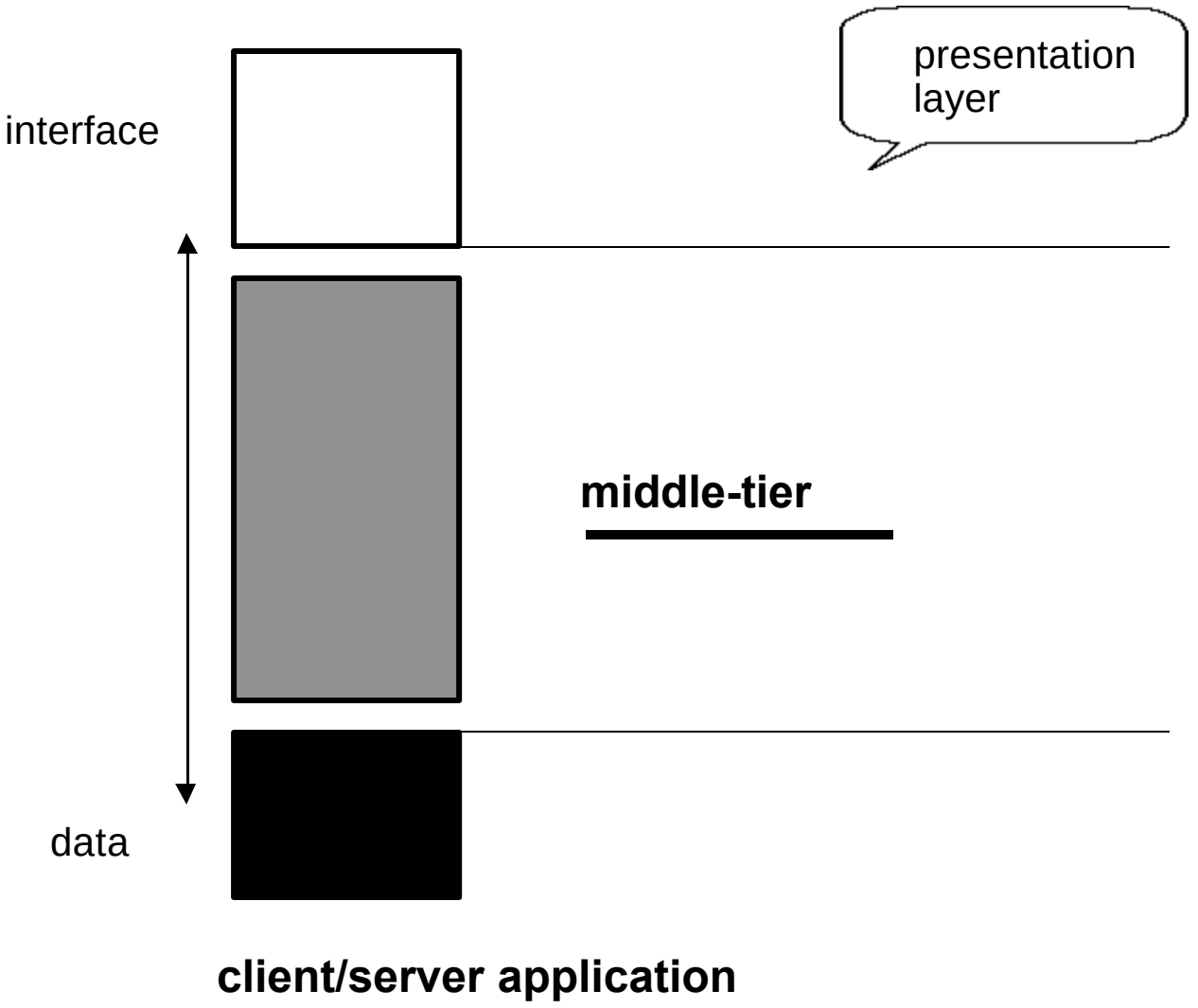
client/server application

From 2-Tier to 3-Tier



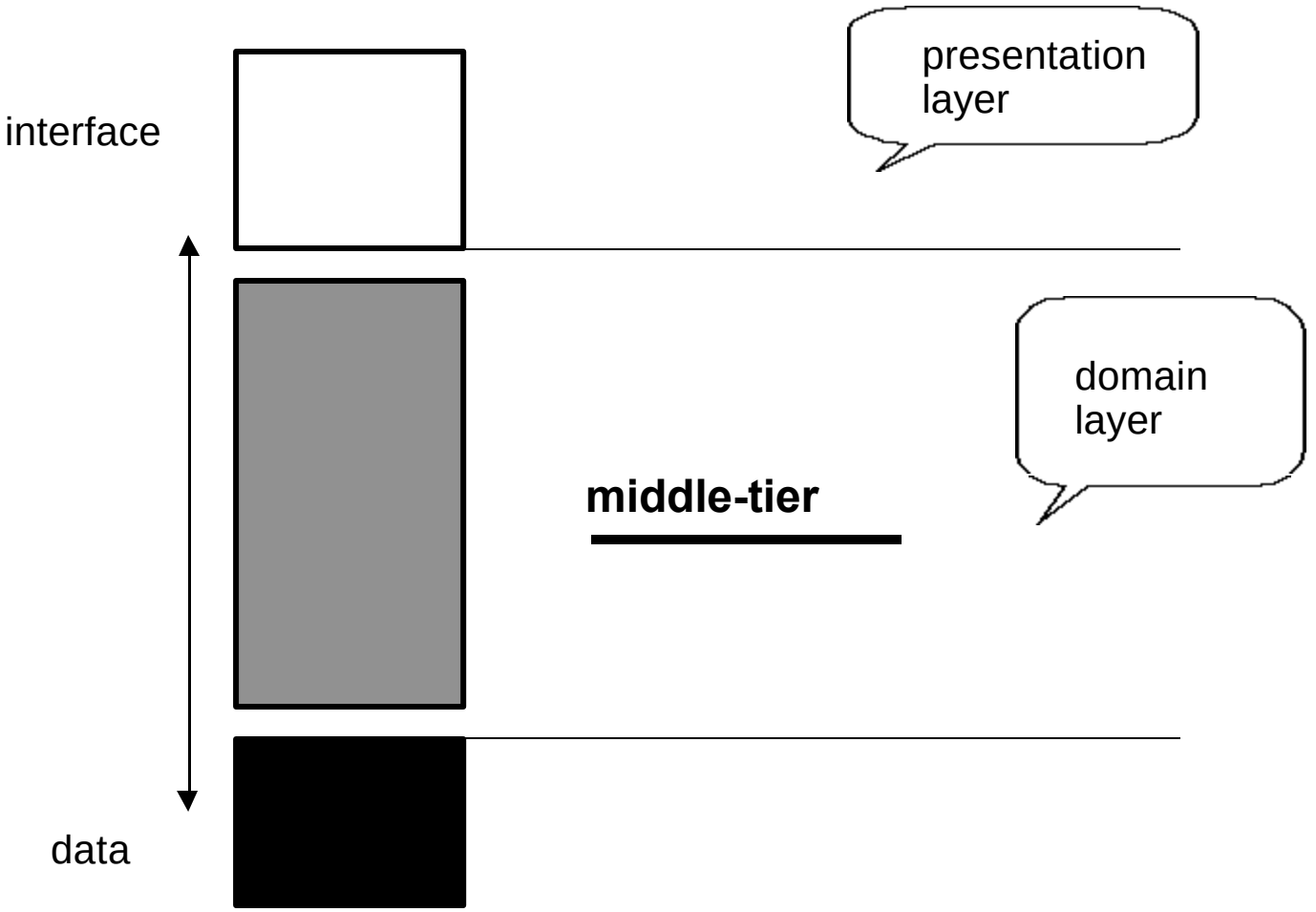
client/server application

From 2-Tier to 3-Tier



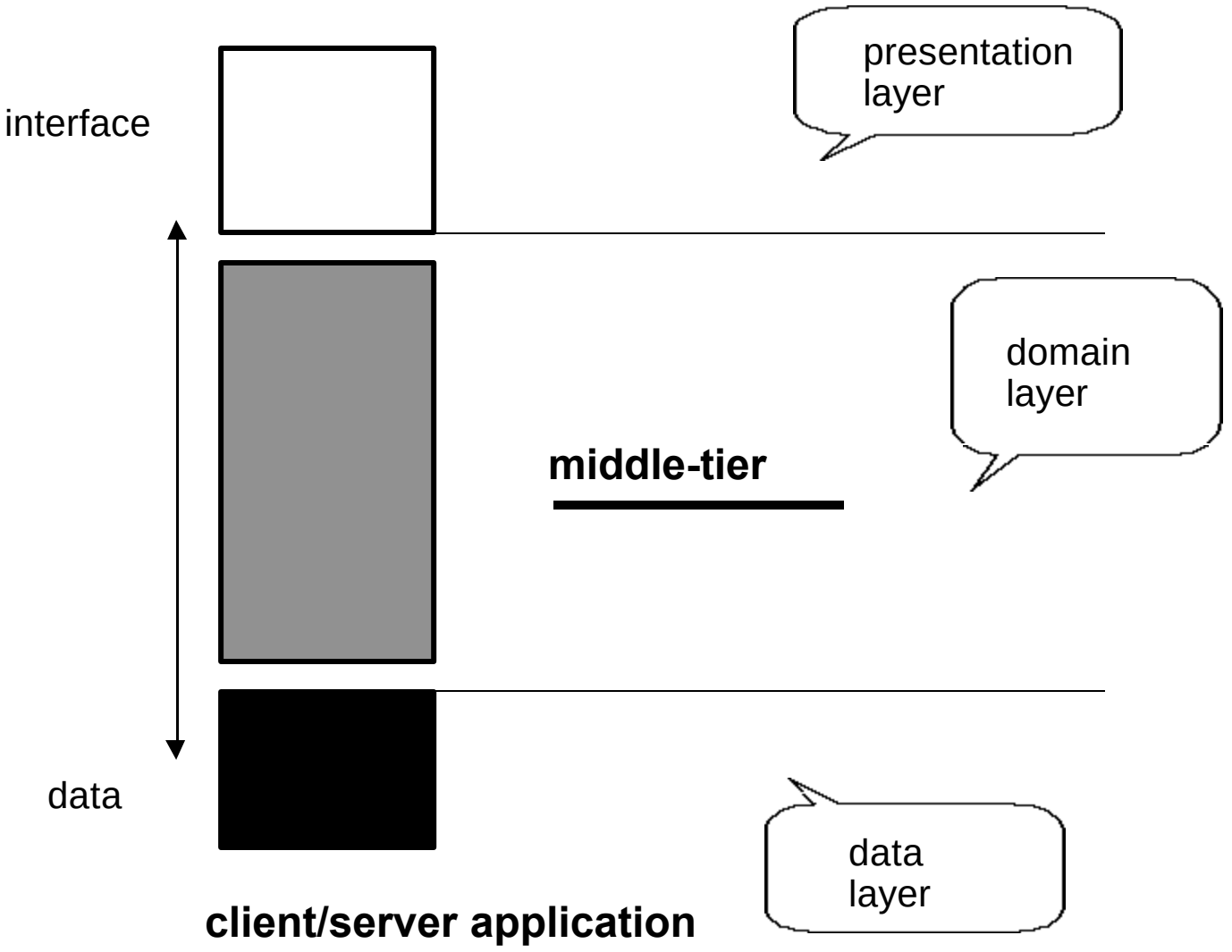
client/server application

From 2-Tier to 3-Tier

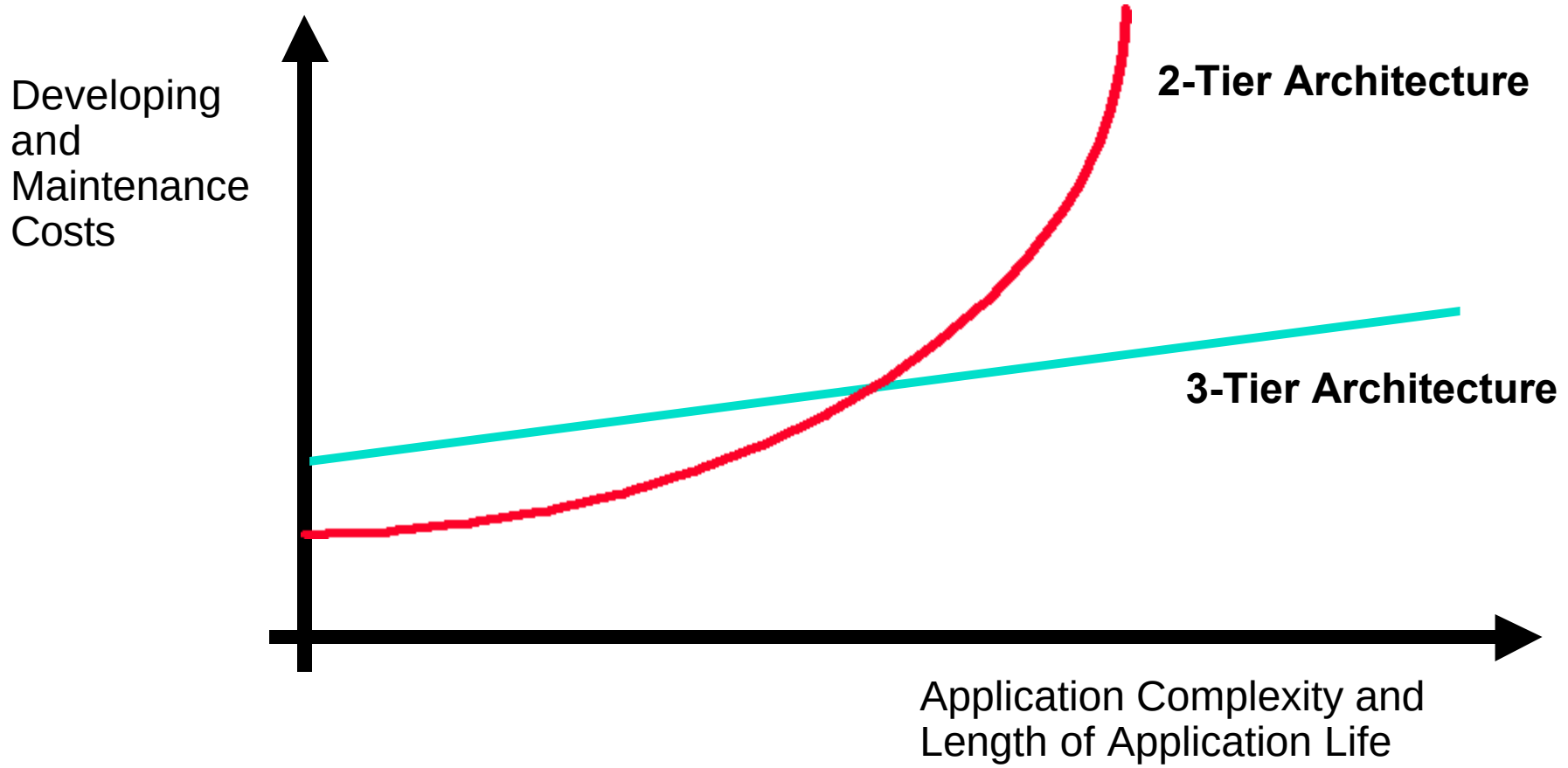


client/server application

From 2-Tier to 3-Tier

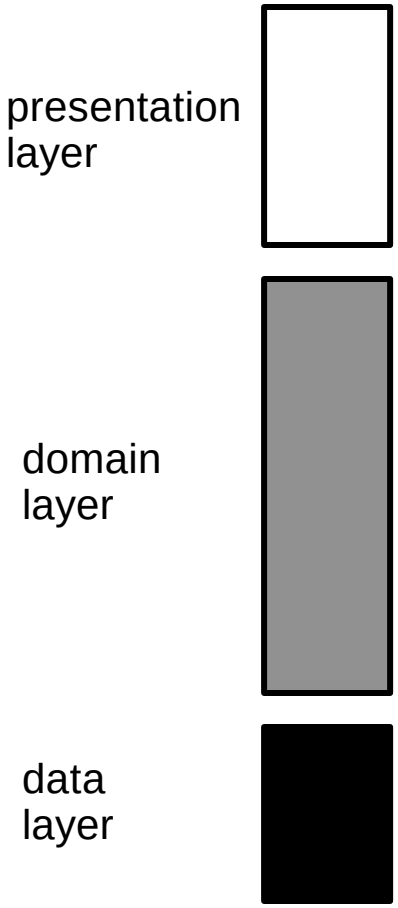


2-Tier and 3-Tier compared



(source: Edwards, 3-Tier Client/Server at Work)

Development for the Web



client/server application

Two classical scenarios

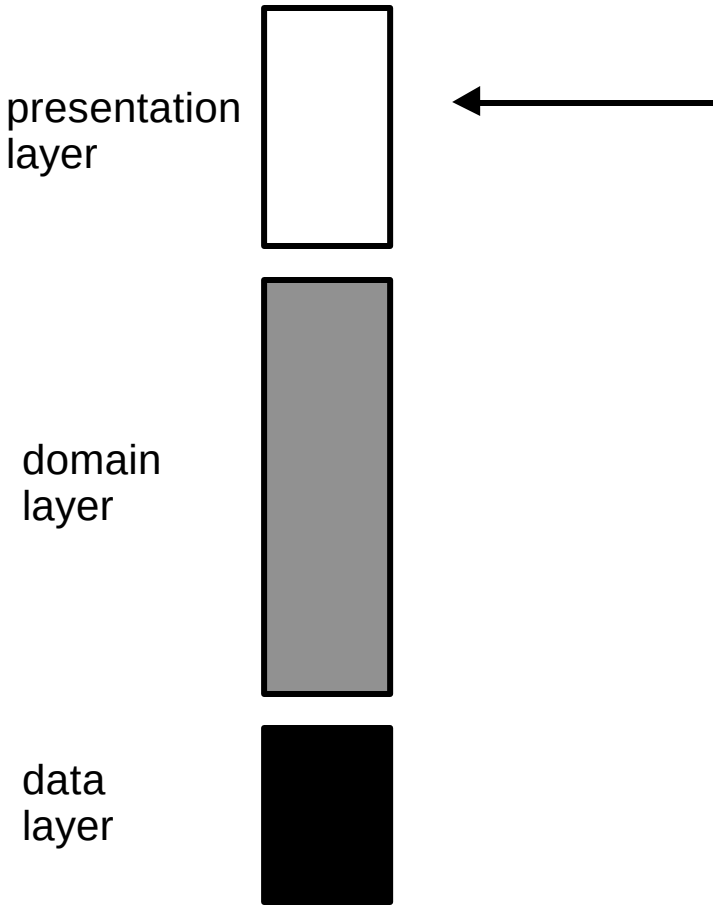
1. Html (+ varoius scripts) + CGI, ASP, ecc.

- “2.5 tier”
- “ad hoc” development

2. Applet (Java)

- Client/server interaction via socket, CGI, Servlet, ASP, etc.
- “ad hoc” development
- “ad hoc” decision how **thin** or how **fat** the client should be

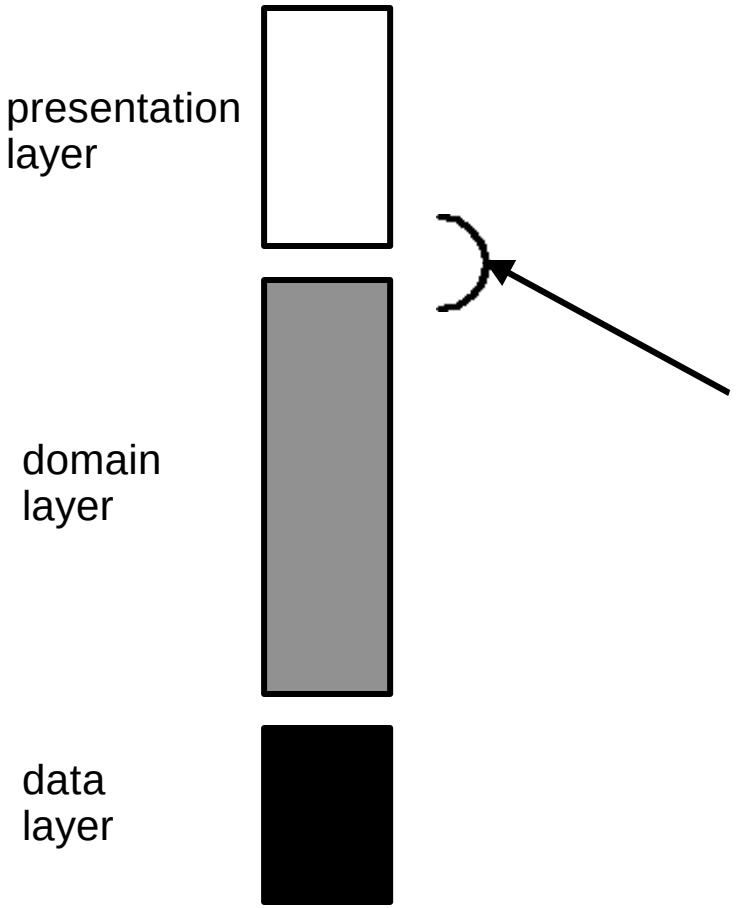
Our approach (1)



- Thin client solution without “ad hoc” decisions during the development

client/server application

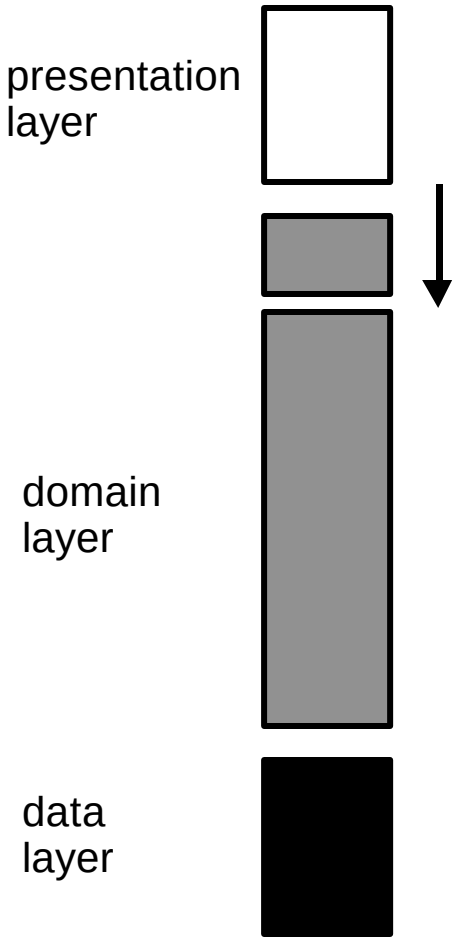
Our approach (2)



- Transparency in the communication between presentation and domain layer during development

client/server application

Our approach (3)

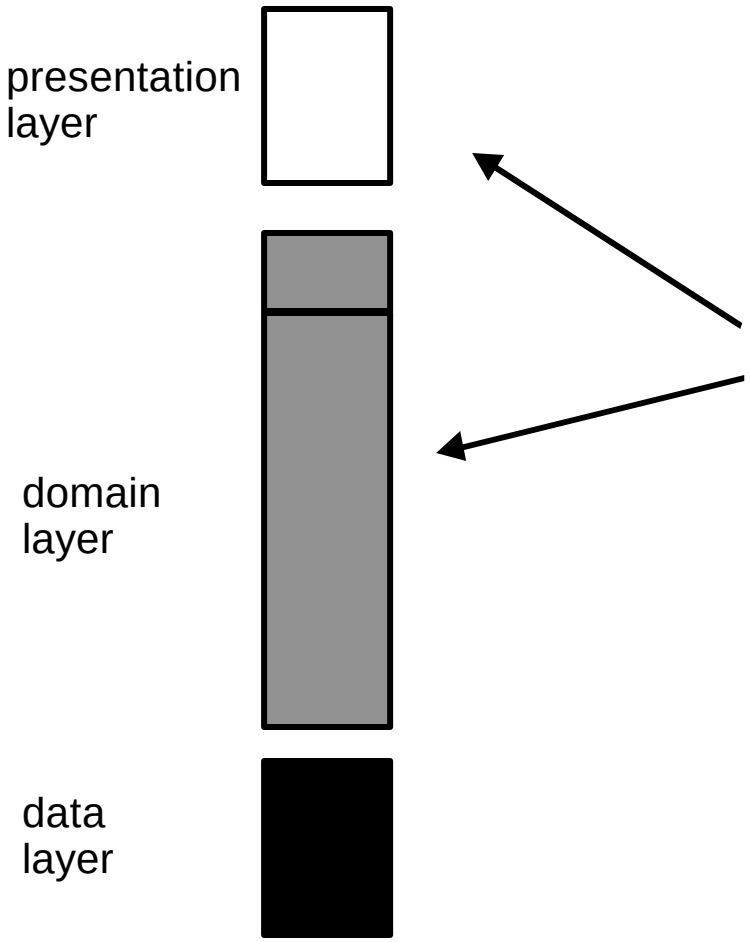


client/server application

- Shift the “Presentation logic sublayer” to the middle tier, further reducing the client size

Presentation logic sublayer
-> composite view controller

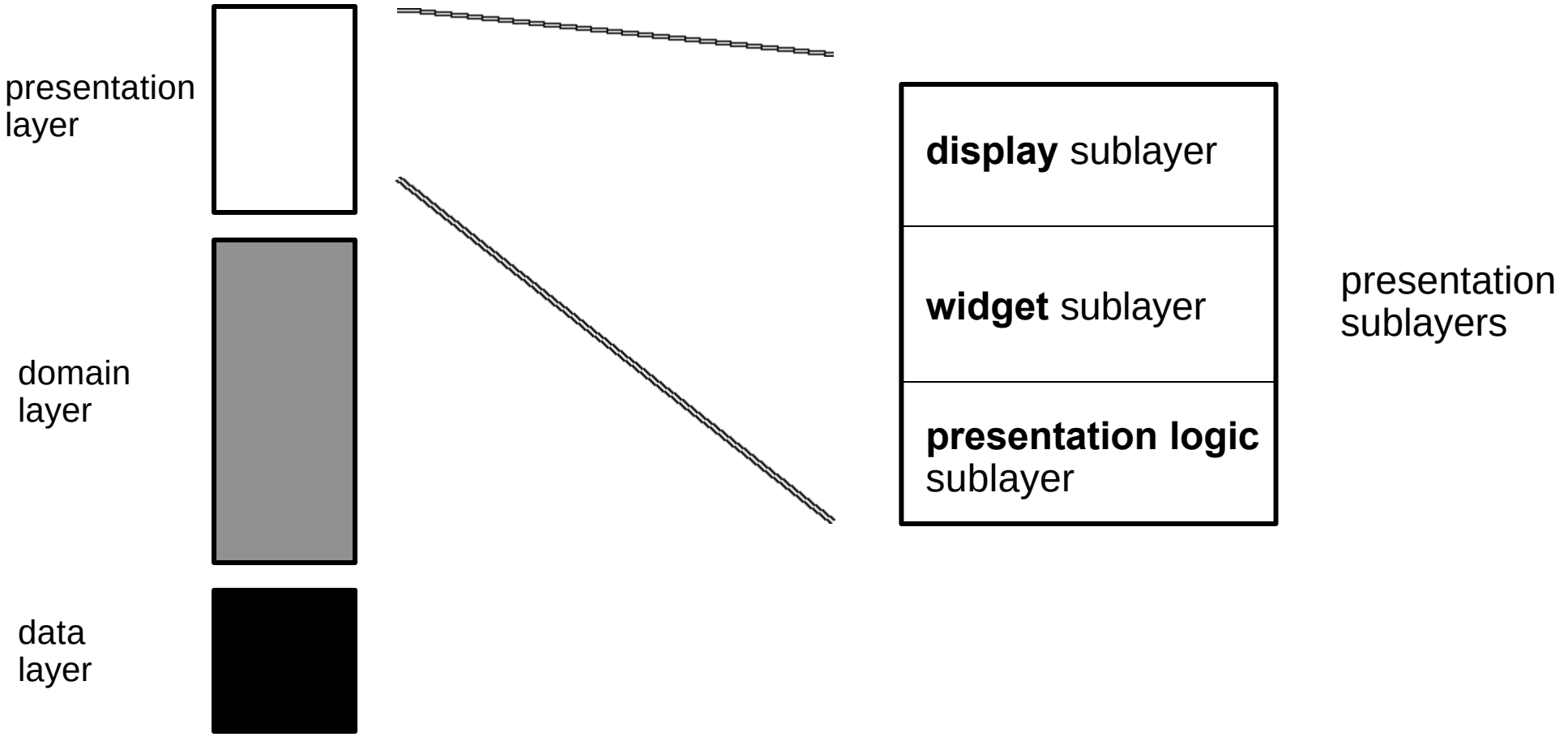
Our approach (4)



- Language independence between presentation and domain layer

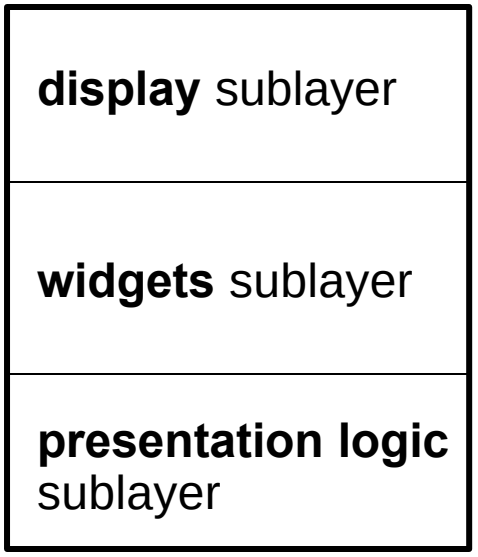
client/server application

Sublayers of the Presentation Layer



client/server application

Sublayers of the Presentation Layer



Interacts directly with the user's IO hardware

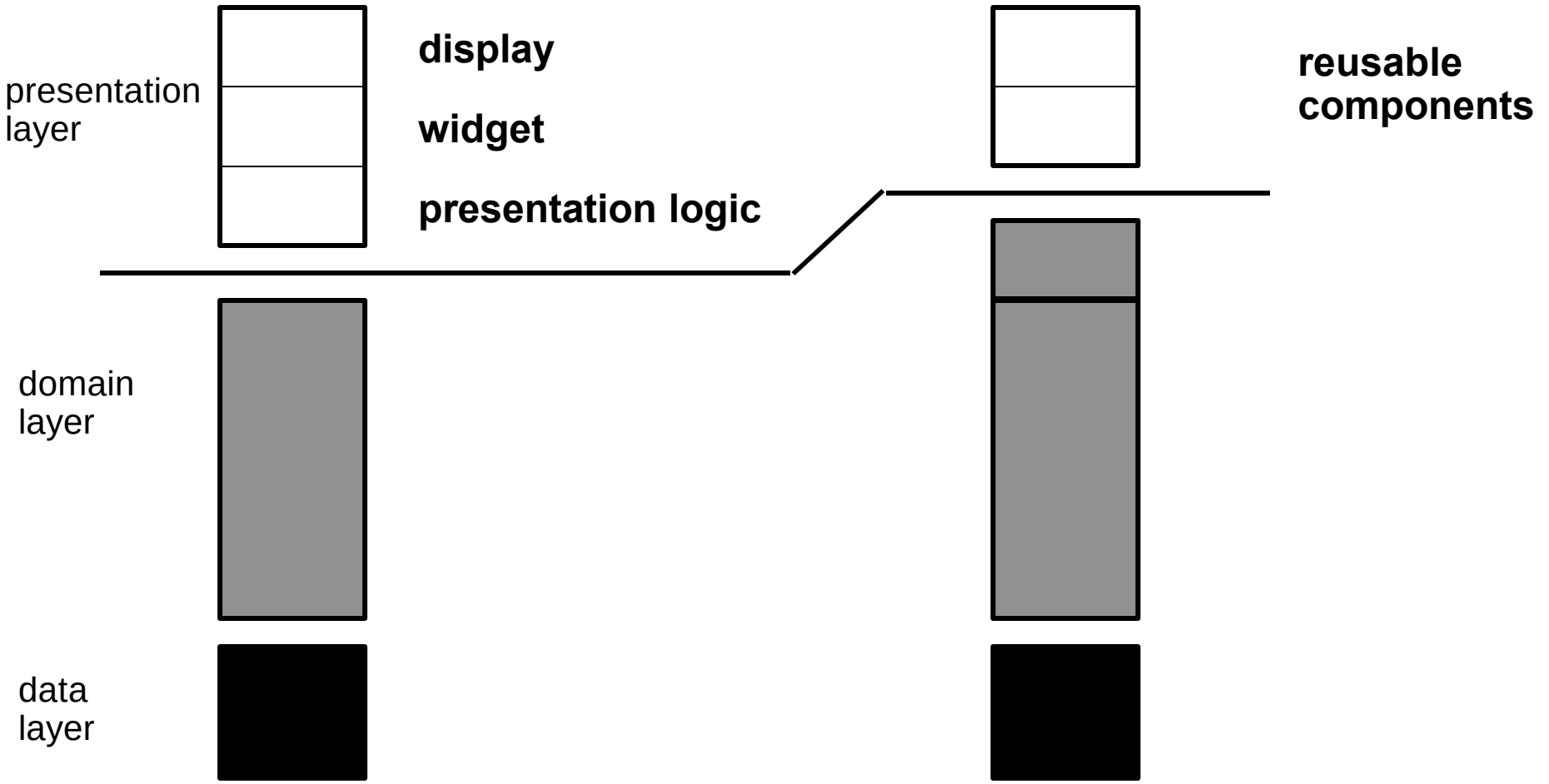
Abstracts the display sublayer offering itself as a collection of objects (buttons, text fields, menues, etc.)

Glues the widgets together making them interact with the remote domain logic layer

Only display and widgets sublayers provide highly reusable components

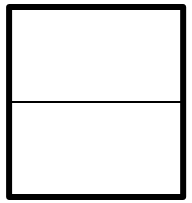
-> set the client/server interface between widgets and presentation logic sublayers

Shifting the presentation logic to the middle-tier



client/server application

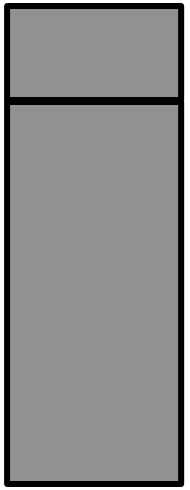
Using Swing for the presentation layer



presentation layer

display

widget



domain layer



data layer

client/server application

- Widget sublayer included

- Full set of portable GUI elements with different “look-and-feels”

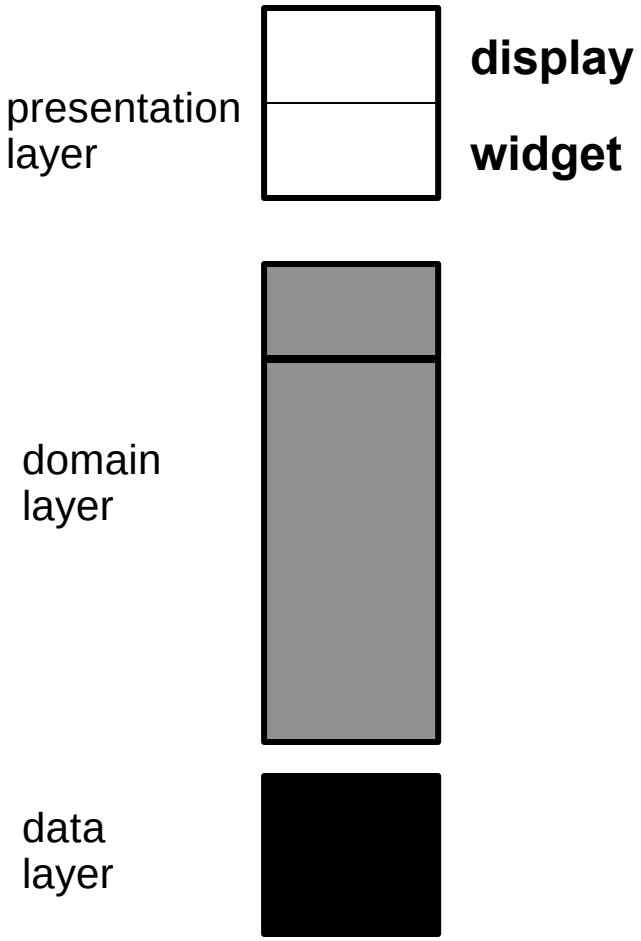
- Included in the client JRE (Java 2)

-> No download at runtime
-> No need of client specific local installations

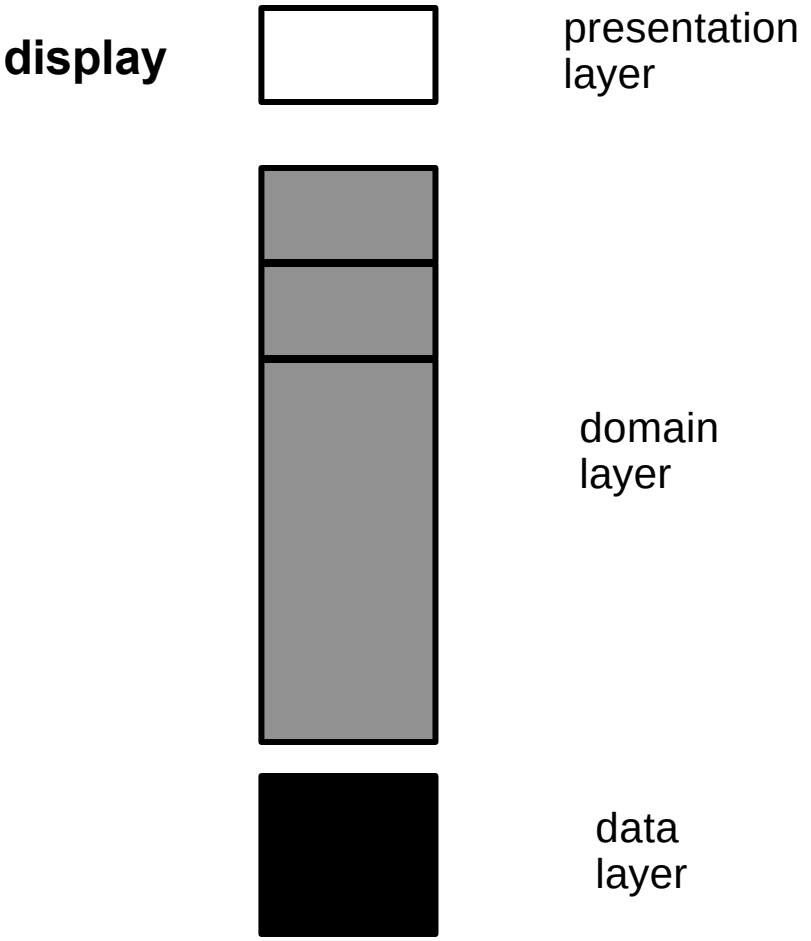
- Delegation model: the reaction on hardware events can be easily forwarded to the next sublayer

-> Event actions run on the domain layer

Comparing with X Window (1)



client/server application



X client/server application

Comparing with X Window (2)

Example: tooltip



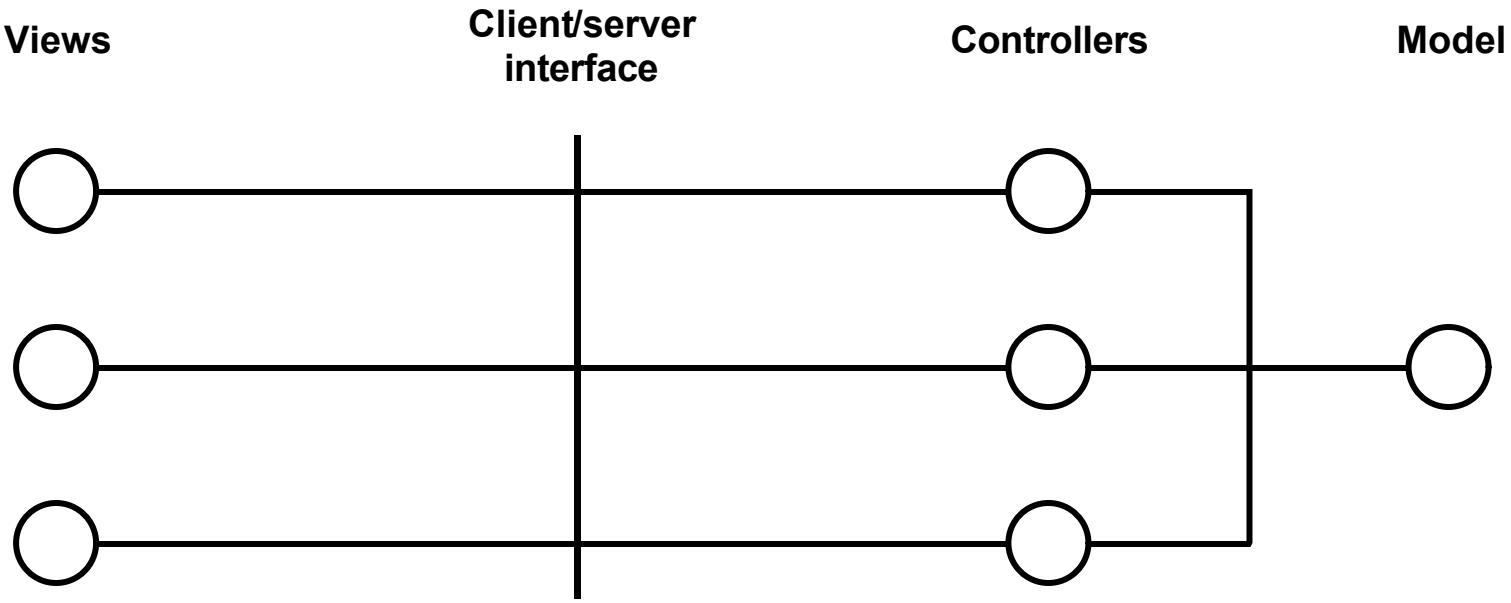
Swing: **local**

X Window: **through the net**

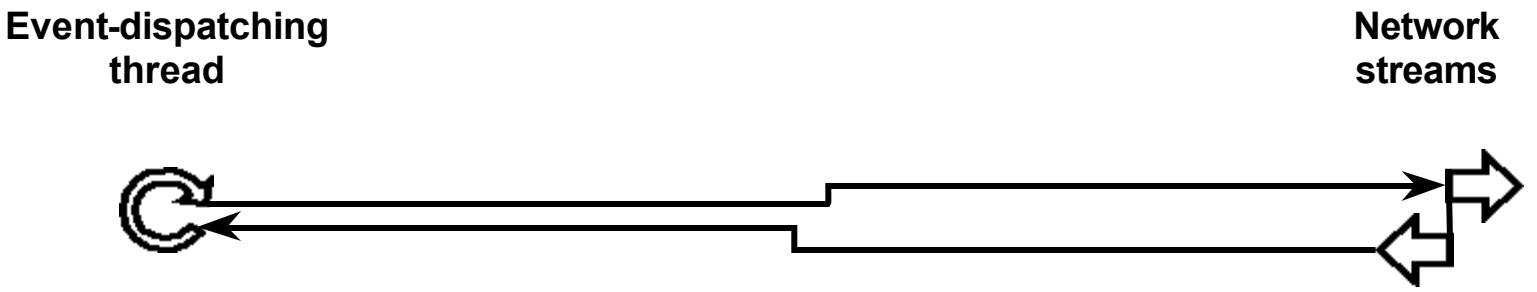
Generic applet

- Setup
- Provides utility classes
  - message, arguments, invocation
- Caching
- Uses reflection mechanisms
- Manages MVC communication

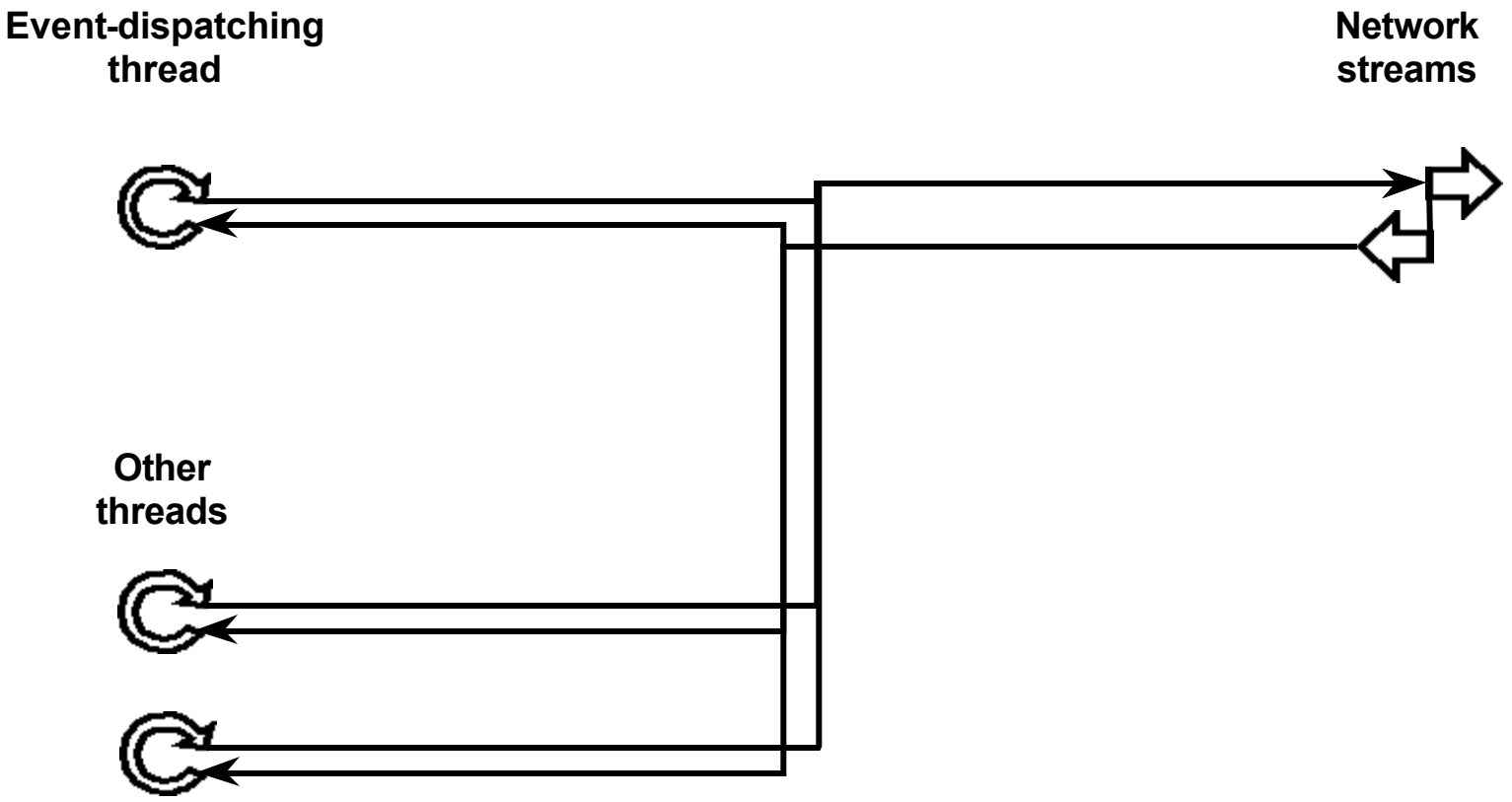
MVC communication



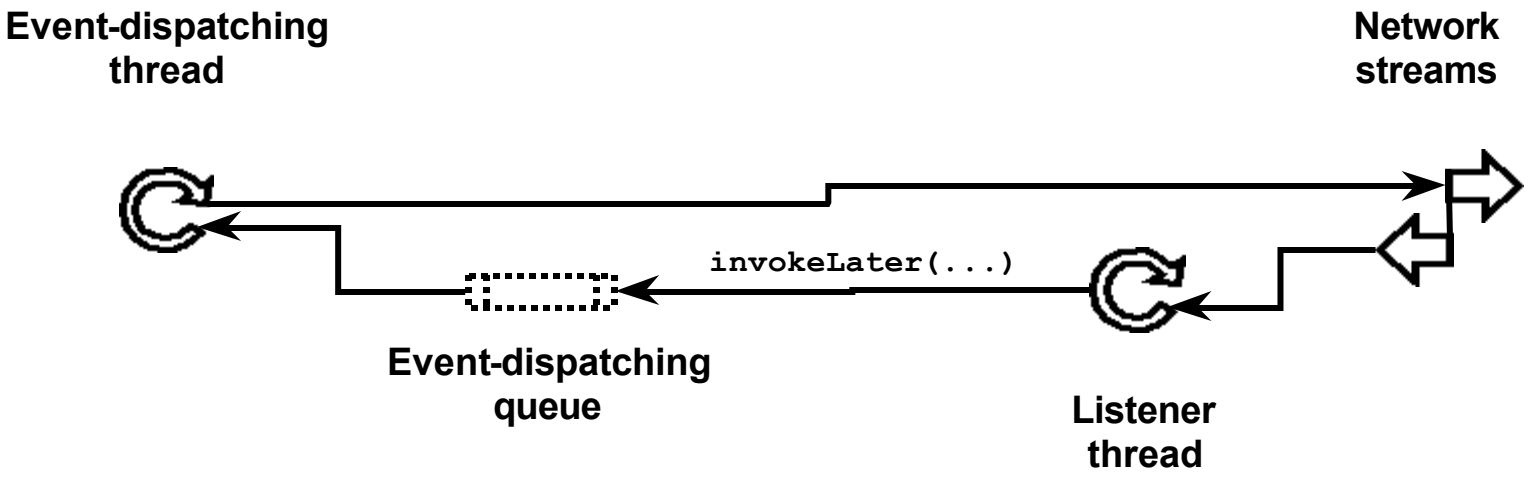
MVC communication
Synchronous unithreaded



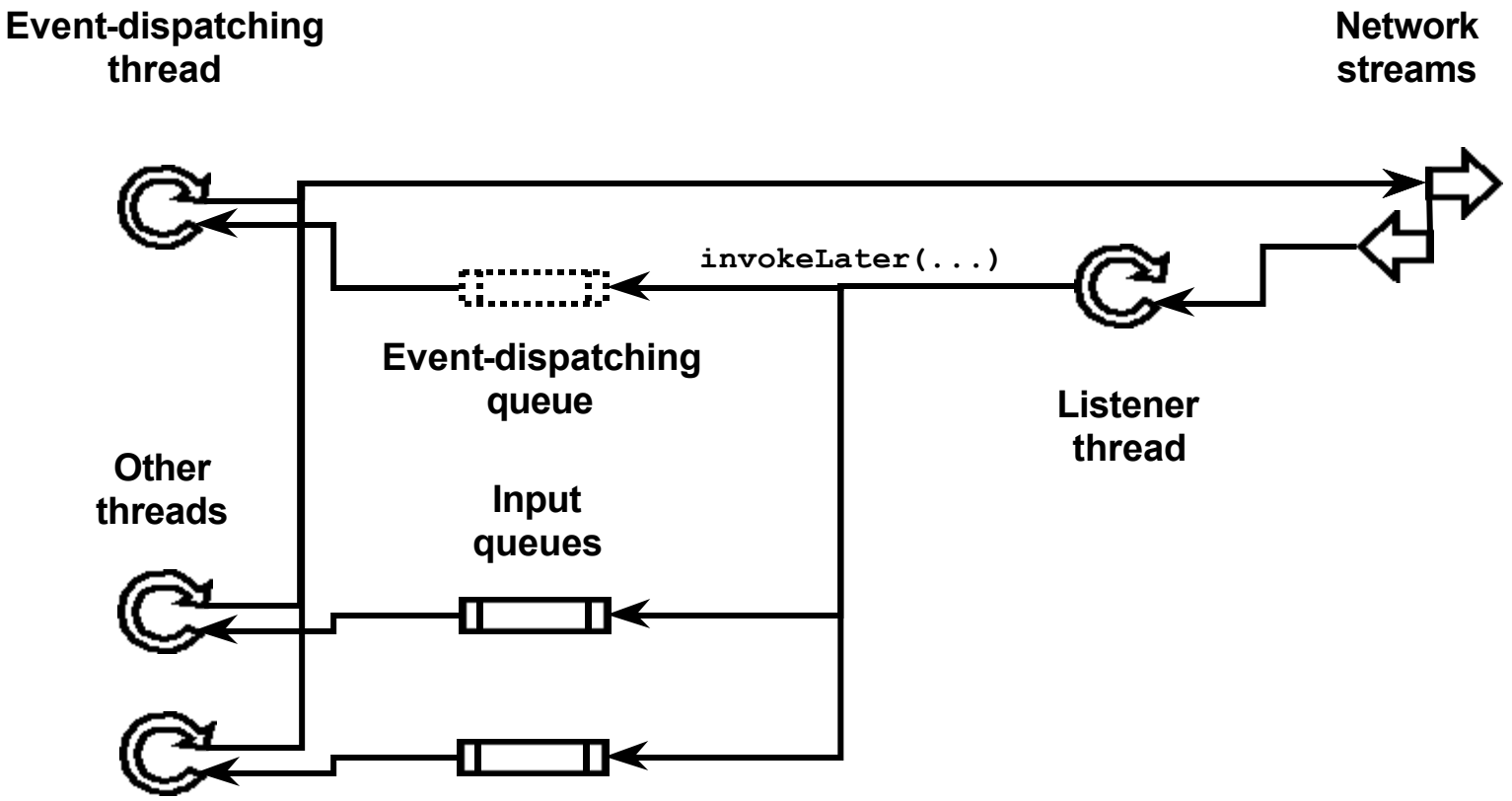
MVC communication
Synchronous multithreaded



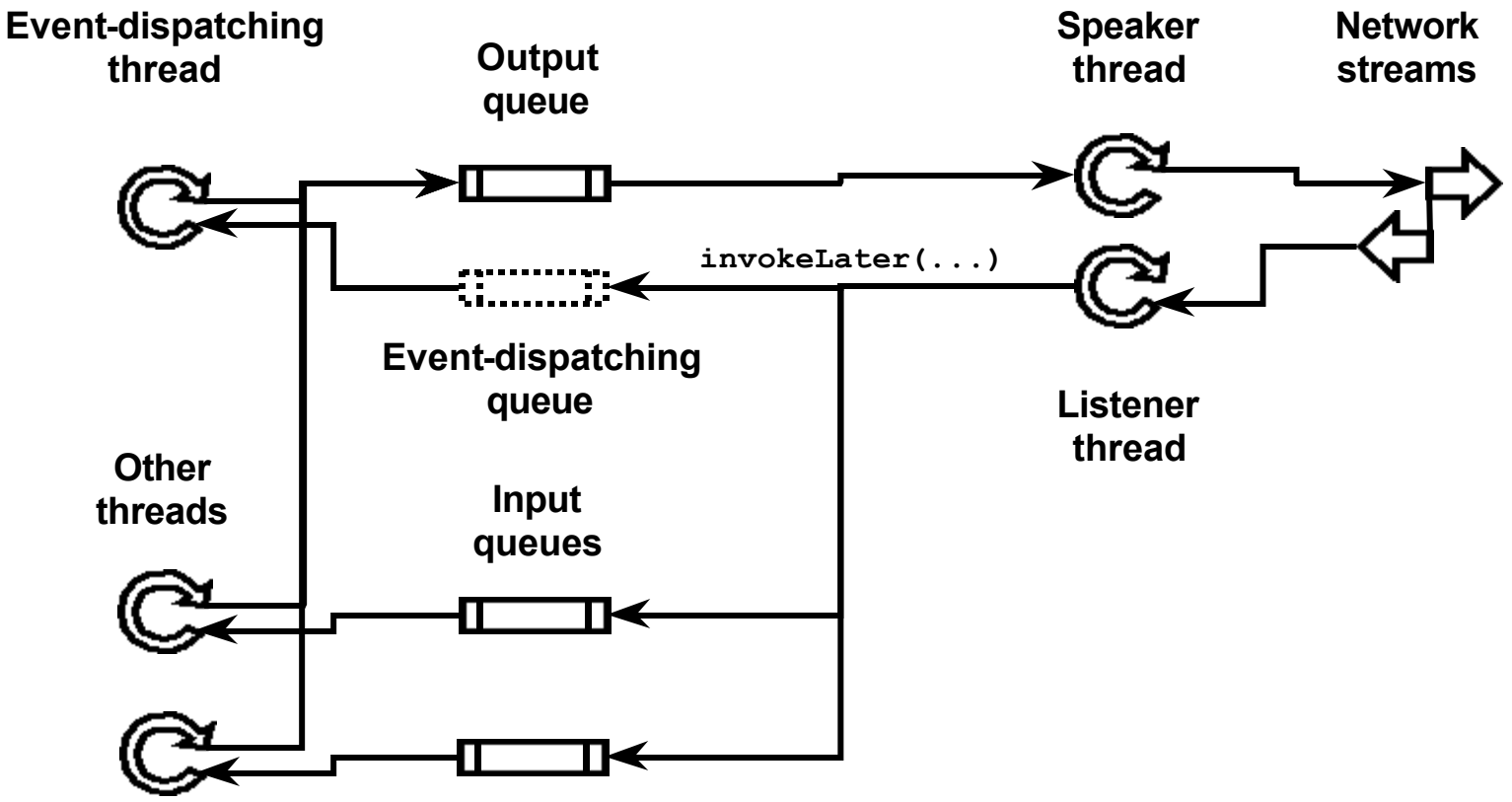
MVC communication
Asynchronous unithreaded



MVC communication
Asynchronous multithreaded (1)



MVC communication
Asynchronous multithreaded (2)



Conclusions

- Solution without “ad hoc” decisions during the development
- Client size reduction
- Transparency in the communication during development
- Language independence between client and middle-tier