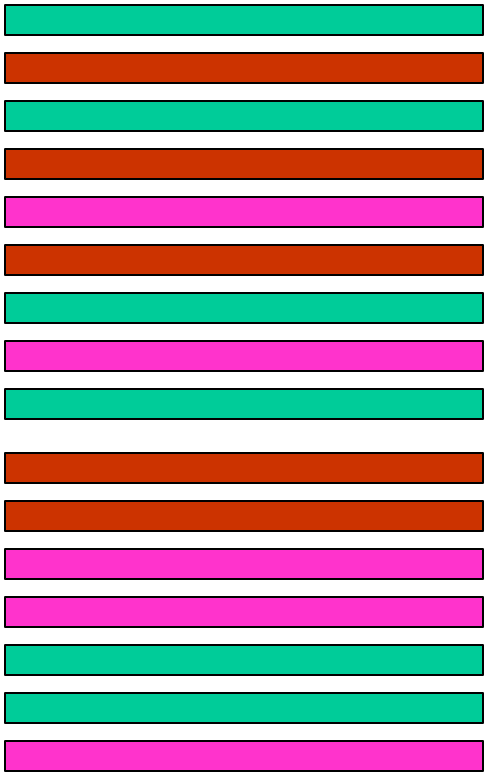
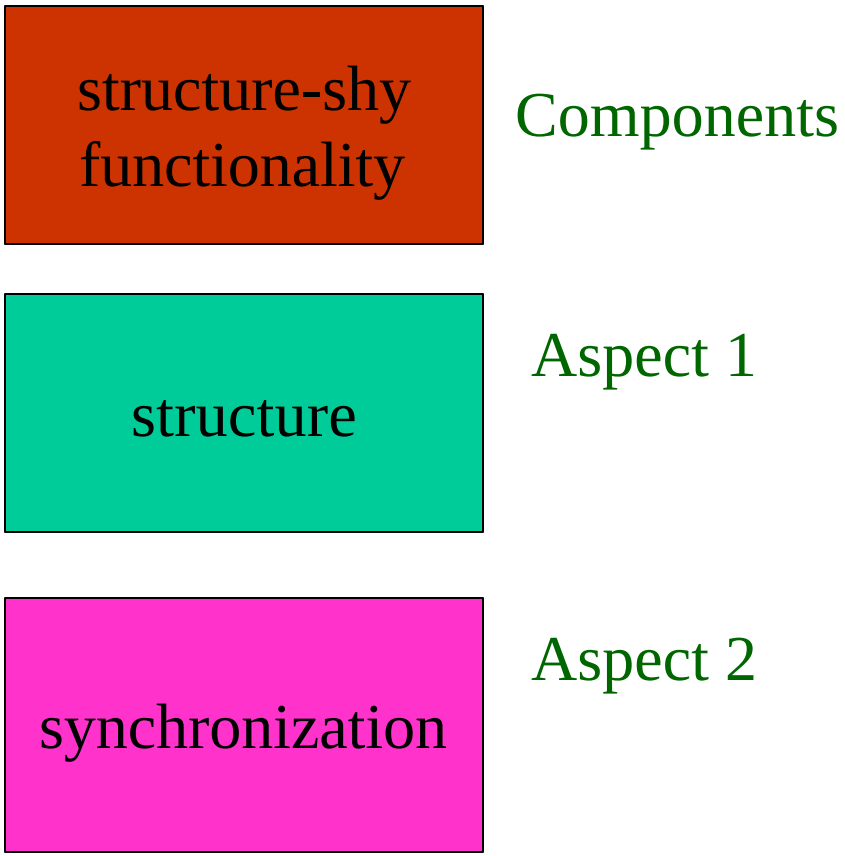
Cross cutting

Cross-cutting of components and aspects

ordinary program



better program



From Crista Lopes PhD thesis (NU/Xerox PARC)

```
interface ShapeI extends Remote {
  double get_x() throws RemoteException ;
  void   set_x(int x) throws RemoteException ;
  double get_y() throws RemoteException ;
  void   set_y(int y) throws RemoteException ;
  double get_width() throws RemoteException ;
  void   set_width(int w) throws RemoteException ;
  double get_height() throws RemoteException ;
  void   set_height(int h) throws RemoteException ;
  void   adjustLocation() throws RemoteException ;
  void   adjustDimensions() throws RemoteException ;
}
public class Shape
       implements ShapeI {
  protected AdjustableLocation loc;
  protected AdjustableDimension dim;
  public Shape() {
    loc = new AdjustableLocation(0, 0);
    dim = new AdjustableDimension(0, 0);
  }
  double get_x() throws RemoteException {
    return loc.x(); }
  void   set_x(int x) throws RemoteException {
    loc.set_x(); }
  double get_y() throws RemoteException {
    return loc.y(); }
  void   set_y(int y) throws RemoteException {
    loc.set_y(); }
  double get_width() throws RemoteException {
    return dim.width(); }
  void   set_width(int w) throws RemoteException {
    dim.set_w(); }
  double get_height() throws RemoteException {
    return dim.height(); }
  void   set_height(int h) throws RemoteException {
     dim.set_h(); }
  void   adjustLocation() throws RemoteException {
    loc.adjust();
  }
  void   adjustDimensions() throws RemoteException {
    dim.adjust();
  }
}
class AdjustableLocation {
  protected double x_,  y_;
  public AdjustableLocation(double x, double y) {
    x_ = x; y_ = y;
  }
  synchronized double get_x() { return x_; }
  synchronized void   set_x(int x) {x_ = x;}
  synchronized double get_y() { return y_; }
  synchronized void   set_y(int y) {y_ = y;}
  synchronized void adjust() {
    x_ = longCalculation1();
    y_ = longCalculation2();
  }
}
class AdjustableDimension {
  protected double width_=0.0, height_=0.0;
  public AdjustableDimension(double h, double w) {
    height_ = h; width_ = w;
  }
  synchronized double get_width() { return width_; }
  synchronized void   set_w(int w) {width_ = w;}
  synchronized double get_height() { return height_; }
  synchronized void   set_h(int h) {height_ = h;}
  synchronized void adjust() {
    width_ = longCalculation3();
    height_ = longCalculation4();
  }
}
```

Instead of writing this

Write this

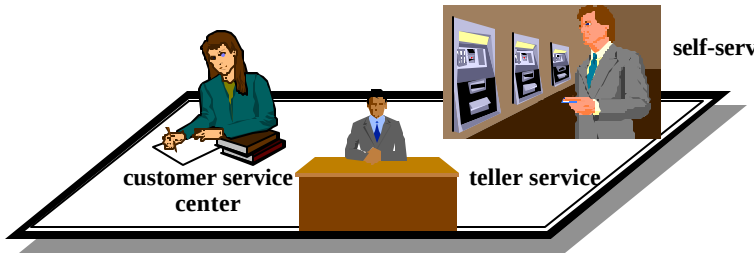
```
public class Shape {
  protected double x_= 0.0,  y_= 0.0;
  protected double width_=0.0, height_=0.0;

  double get_x() { return x_(); }
  void   set_x(int x) { x_ = x; }
  double get_y() { return y_(); }
  void   set_y(int y) { y_ = y; }
  double get_width(){ return width_(); }
  void   set_width(int w) { width_ = w; }
  double get_height(){ return height_(); }
  void   set_height(int h) { height_ = h; }
  void   adjustLocation() {
    x_ = longCalculation1();
    y_ = longCalculation2();
  }
  void   adjustDimensions() {
    width_ = longCalculation3();
    height_ = longCalculation4();
  }
}
```

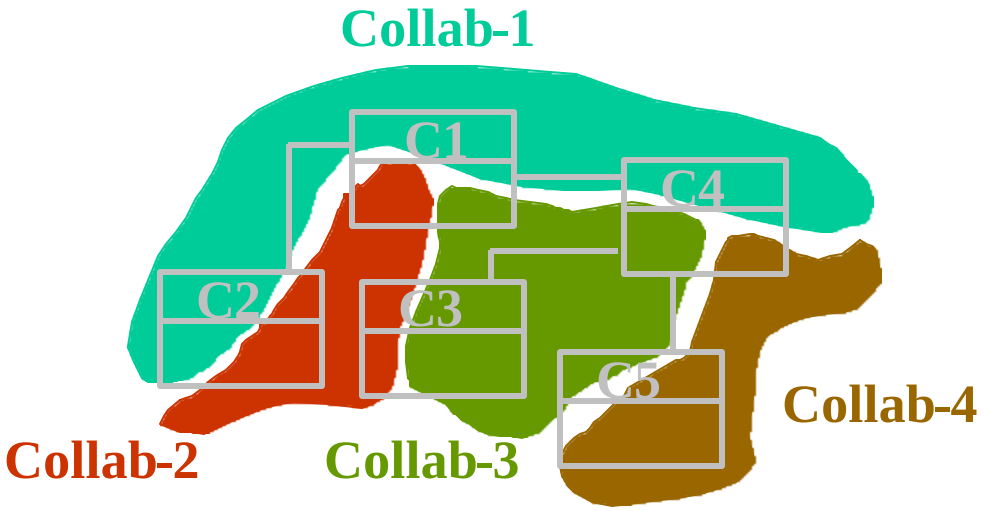
```
coordinator Shape {
  selfex adjustLocation, adjustDimensions;
  mutex {adjustLocation, get_x, set_x,
                         get_y, set_y};
  mutex {adjustDimensions, get_width, get_height,
                           set_width, set_height};
}
```

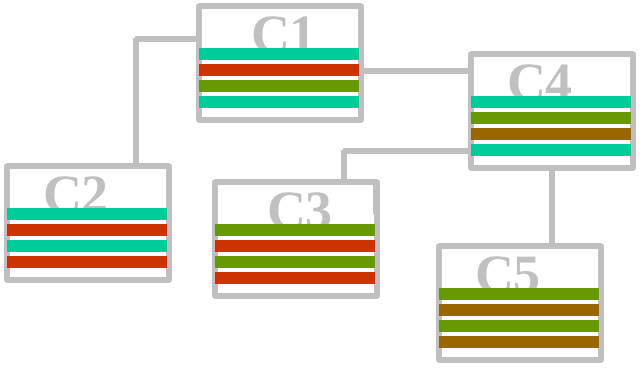
```
portal Shape {
  double get_x() {} ;
  void   set_x(int x) {};
  double get_y() {};
  void   set_y(int y) {};
  double get_width() {};
  void   set_width(int w) {};
  double get_height() {};
  void   set_height(int h) {};
  void   adjustLocation() {};
  void   adjustDimensions() {};
}
```

What's the problem? TANGLING



OOAD





Implementation

Reconciliation of Both Worlds: Adaptive Plug-n-Play Components (APPC)