

PROGRAMMERS
**RIDING
THE IT
CREST**

SECRETS OF THE
ROCK STAR
PROGRAMMERS®

ED BURNS



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private boolean isTreeTraversalCallback() { public boolean
takeAction(FacesContext context, UIComponent comp, boolean isTreeTraversalCallback) { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

AGENDA

🌀 Introduction

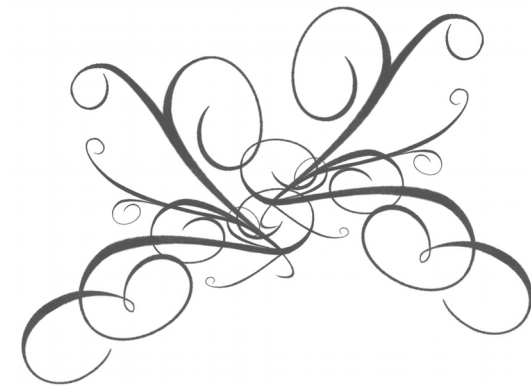
- ★ Why I wrote the book
- ★ What's my idea of a “Rock Star Programmer”
- ★ Prior art

🌀 Categories of Secrets

- ★ What do I mean by “Secret”?
- ★ The problem with categories

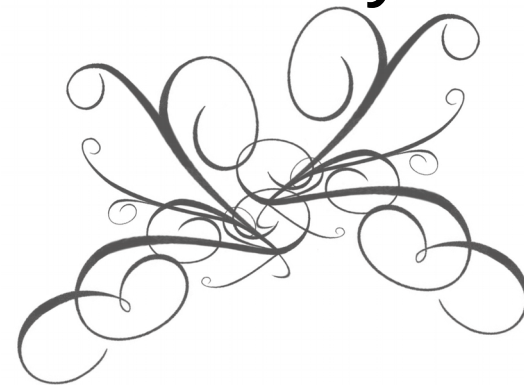
🌀 Ancient Secrets

🌀 Modern Secrets



WHY I WROTE THIS BOOK

- ❧ I started speaking at conferences, often the same speakers again and again. Why?
- ❧ A friend wrote an interview book, it seemed like a good way to explore the above question.
- ❧ I work from home. So my cubicle mates could be anywhere. What if could ask top programmers questions as if they were my cubicle mates?



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



SO, WHAT IS A ROCK STAR PROGRAMMER?

- ☿ It's totally subjective
- ☿ A good mix of all the skills
- ☿ Not a jerk
- ☿ Someone you'd work really hard to hire onto your team
- ☿ Someone whose team you'd work really hard to join



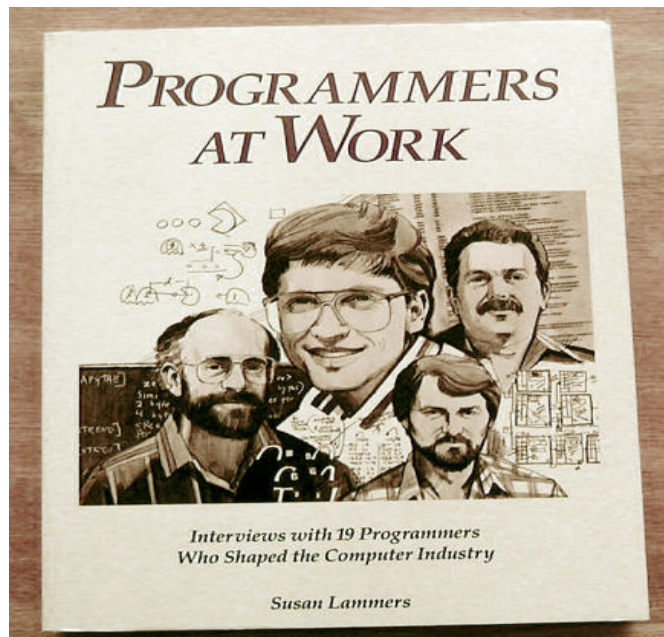
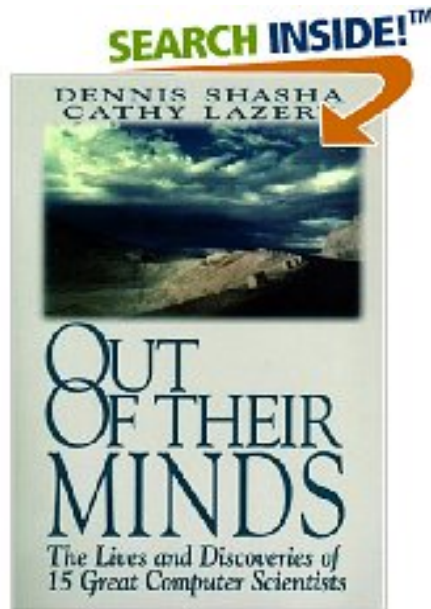
```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate = false;
private boolean isImmediateCallback() { public boolean
takeAction(FacesContext context, UIComponent comp, ActionListener listener) { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

PRIOR ART

NO ONE IS THE FIRST TO DO ANYTHING

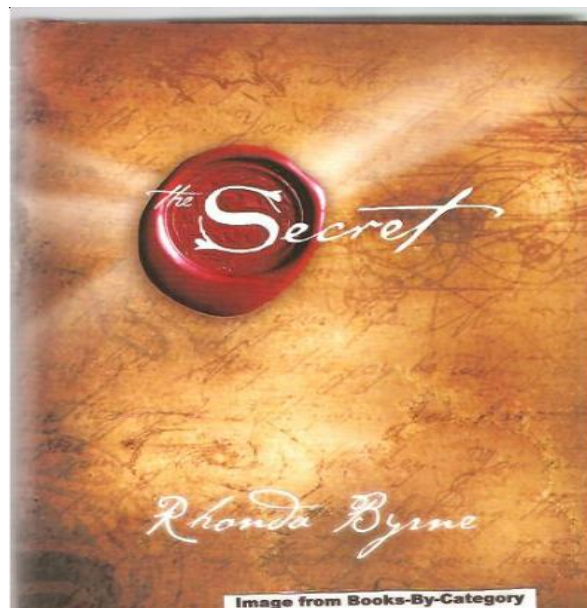
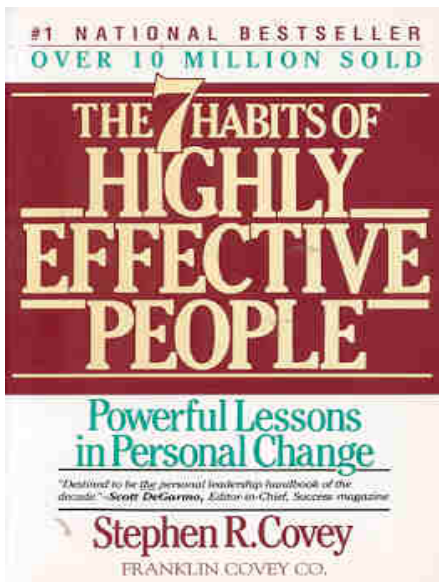
☿ Susan Lammers: Programmers at Work

☿ Shasha, Lazere: Out of Their Minds



WHAT IS A "SECRET"?

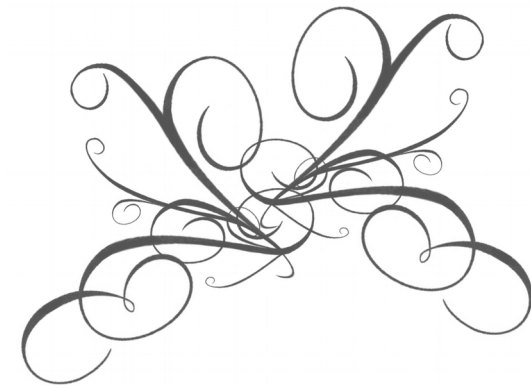
- ❧ Character Attribute
- ❧ Habit (as in *The Seven Habits of Highly Effective People* by Steven Covey)
- ❧ *The Secret* by Rhonda Byrne



Ancient Secrets



Modern Secrets



BEWARE OF CATEGORIZATION

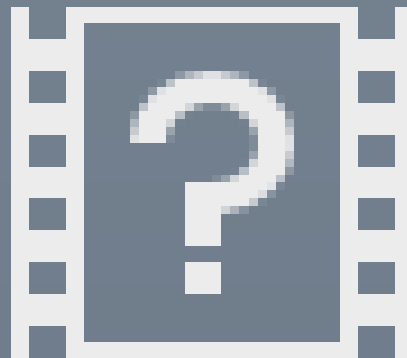
James Gosling is wary of categorization



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private FacesContext context;
private boolean isImmediate;
private boolean isPartialCallback() { public boolean
takeAction(FacesContext context, UIComponent comp, ActionSource as) { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

CATEGORIES

BEWARE OF CATEGORIZATION

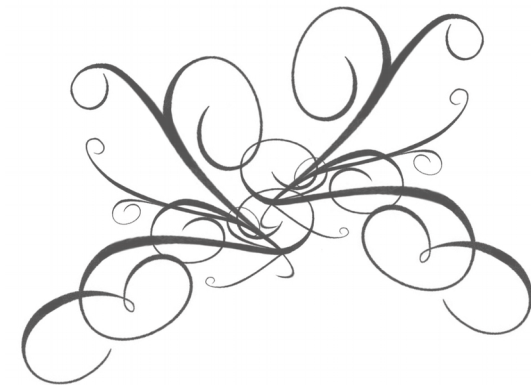
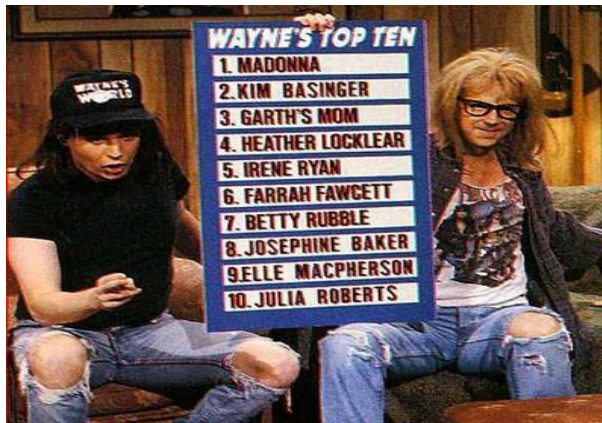



```
public class AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private boolean takeAction = false;
private FacesContext context;
private UIComponent component;
private boolean isComp instanceof ActionSource) {
ActionSource comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

CATEGORIES

BEWARE OF CATEGORIZATION

James Gosling is wary of categorization



BEWARE OF CATEGORIZATION

 Adrian Colyer thinks it's useful

Aspect Oriented Programming Pioneer



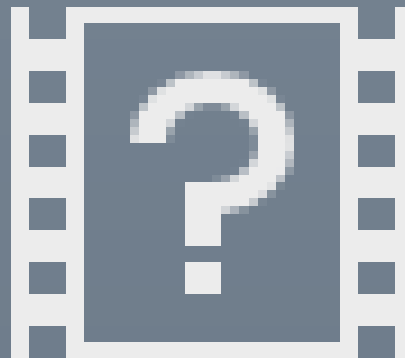
 Venture partner at Accel



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private FacesContext context;
private boolean isImmediate;
private boolean isPartialCallback() { public boolean
takeAction(FacesContext context, UIComponent comp, ActionSource as) { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

CATEGORIES

BEWARE OF CATEGORIZATION



HUMILITY AND CELEBRITY

James Gosling on being known as “The Father of Java”



ANCIENT SECRETS

☿ Things that have been true since at least the dawn of western civilization

- ★ Balance between humility and pride
- ★ Awareness of one's own ignorance
- ★ The Importance of Collaboration
- ★ The Invisible College
- ★ Luck



ANCIENT SECRET

PRIDE TEMPERED BY HUMILITY

☿ The dangers of hubris:
Achilles



☿ Pride goes before a fall

☿ Rabbi Simcha Bunim:

“Every person should have two pockets. In one, there should be a note that says ‘for my sake was the world created.’ In the second, there should be a note that says, ‘I am dust and ashes.’”

https://en.wikipedia.org/wiki/Simcha_Bunim_of_Peshischa

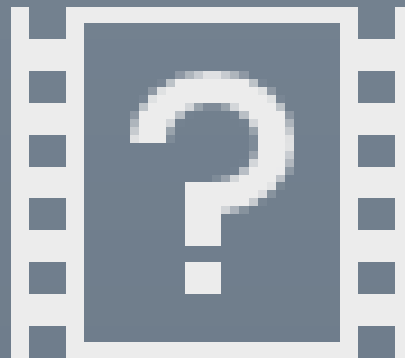
ANCIENT SECRET

PRIDE TEMPERED BY HUMILITY

🌀 Rod Johnson, creator of the Spring Framework

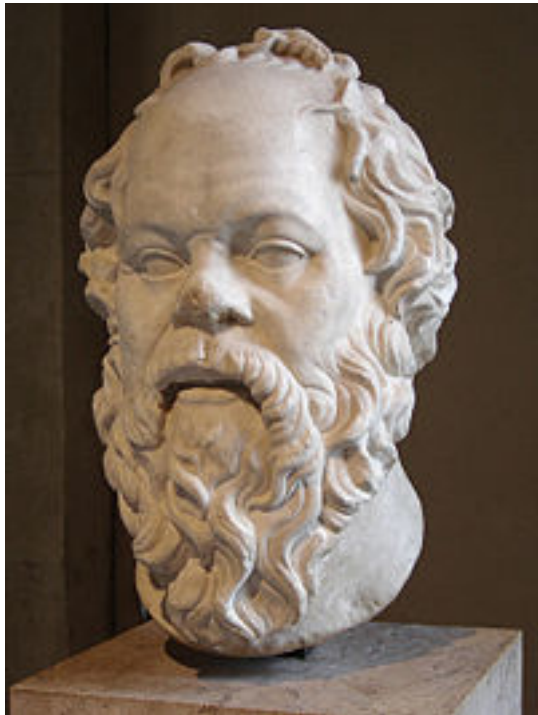



PRIDE TEMPERED BY HUMILITY



ANCIENT SECRET

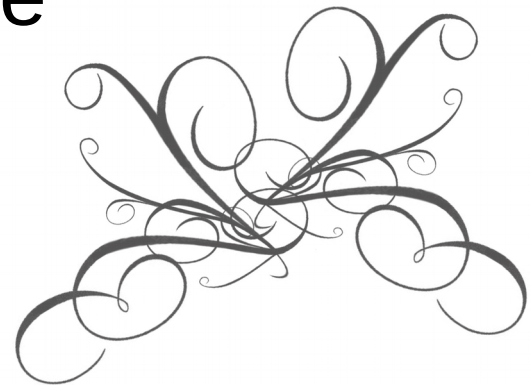
AWARENESS OF YOUR IGNORANCE





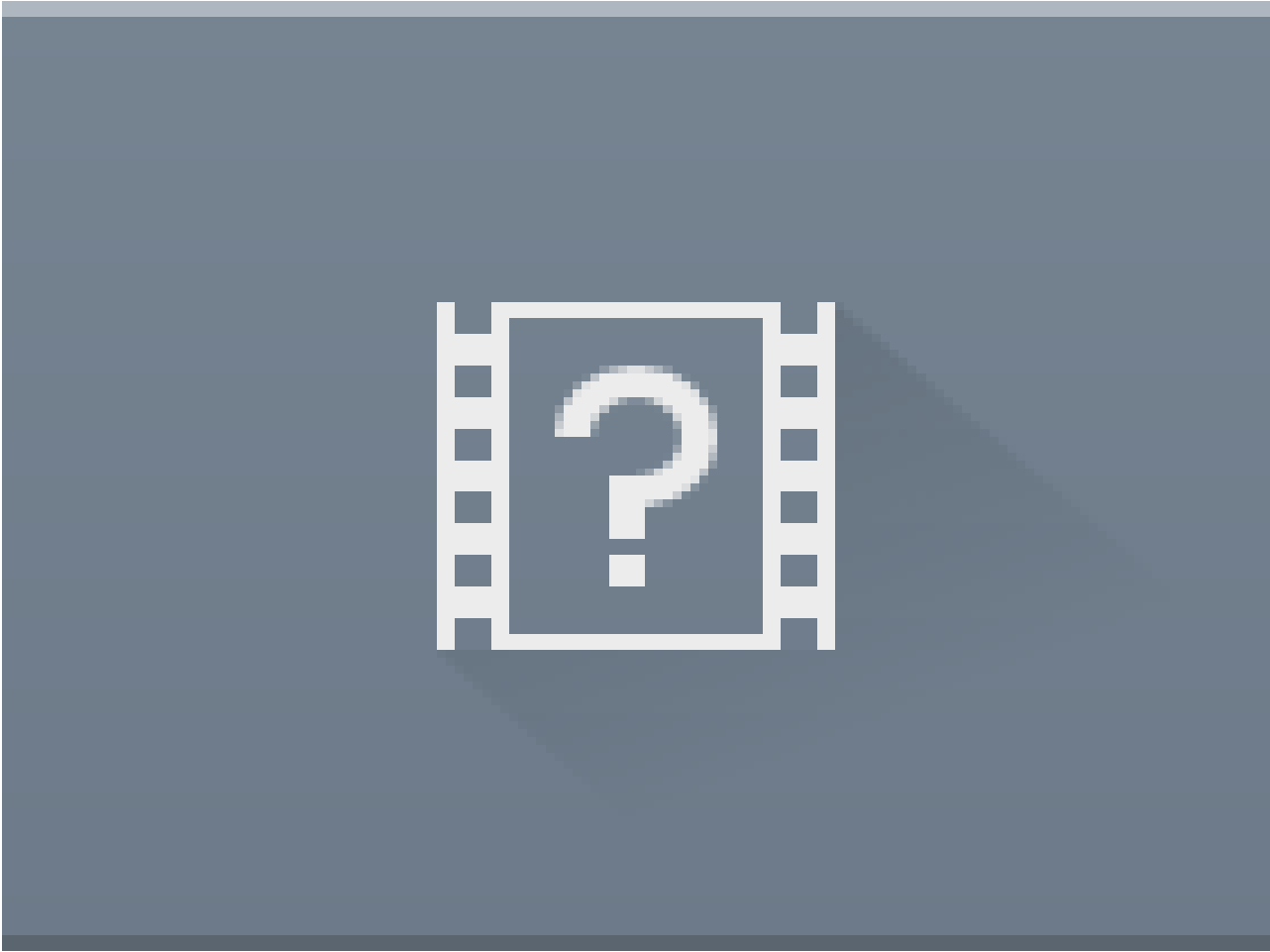
The Orders of Ignorance

- ☿ Phillip Armour, October 2000 CACM
- ☿ 0th Order Ignorance: Lack of Ignorance
- ☿ 1st Order Ignorance: Lack of Knowledge
- ☿ 2nd Order Ignorance: Lack of Awareness
- ☿ 3rd Order Ignorance: Lack of a *Suitably Efficient* process to get to 2OI
- ☿ 4th Order Ignorance: Meta-Ignorance



ANCIENT SECRET

AWARENESS OF YOUR IGNORANCE



ANCIENT SECRET

AWARENESS OF YOUR IGNORANCE



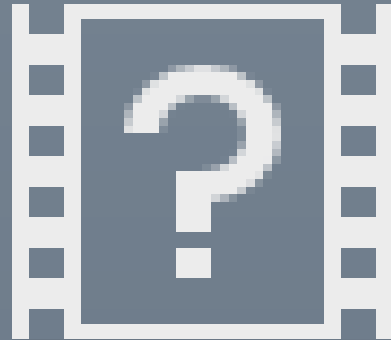
🌀 Courage in the face of other's expertise

🌀 Willingness and ability to share your own expertise



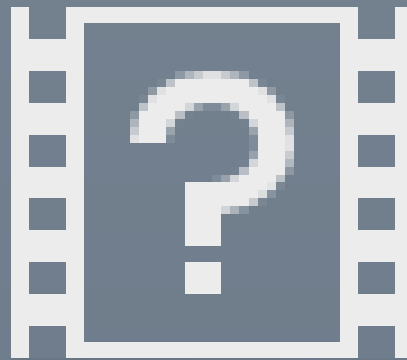
```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> children = new ArrayList<UIComponent>();
public boolean takeAction(FacesContext ctx, UIComponent comp, throws FacesException) throws ActionSource { public boolean
takeAction(FacesContext ctx, UIComponent comp, throws FacesException) throws ActionSource {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

AWARENESS OF YOUR IGNORANCE



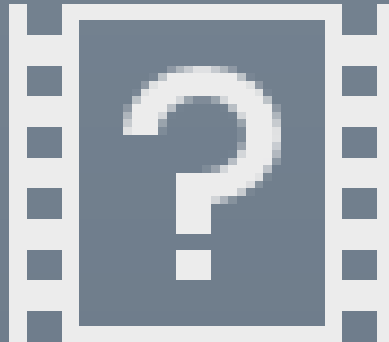

```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate() { public boolean
takeAction(FacesContext ctx, UIComponent comp) throws FacesException { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

AWARENESS OF YOUR IGNORANCE




```
public class AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate = false;
private FacesContext facesContext;
private UIComponent component;
private boolean isImmediate() { public boolean
takeAction() {
if (facesContext.isActionSource(component)) {
throw new FacesException("Component is not an instance of ActionSource");
}
if (component instanceof ActionSource) {
ActionSource as = (ActionSource) component;
if (!as.isImmediate()) {
as.setImmediate(true);
modifiedComponents.add(component);
}
}
else if
(component instanceof EditableValueHolder) {
EditableValueHolder ev = (EditableValueHolder) component;
if (!ev.isImmediate()) {
ev.setImmediate(true);
modifiedComponents.add(component);
}
}
return true;
}
}
```

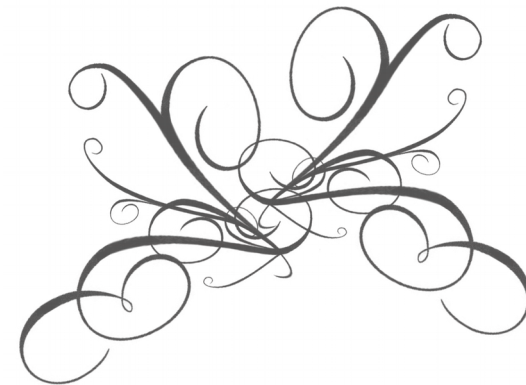
QUESTION CONVENTIONAL WISDOM



ANCIENT SECRET

COLLABORATION

- 🌀 Odysseus and his crew get back home
- 🌀 Aeneas and his survivors found Rome



ANCIENT SECRET

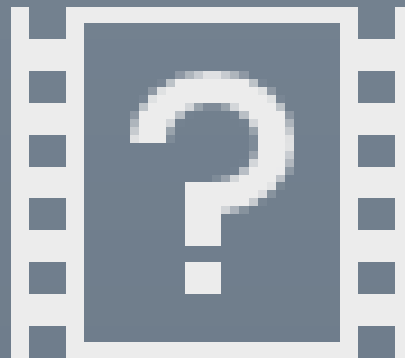
COLLABORATION



🌀 Andy Hunt, co-founder of the Pragmatic Programmers



COLLABORATION





```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate() { public boolean
takeAction(FacesContext ctx, UIComponent comp, ActionListener listener) throws FacesException { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

ANCIENT SECRET

THE INVISIBLE COLLEGE

🌀 Tyco Brahe, Johannes Kepler & co:
annotations in books

🌀 Unfiltered, informal communications produced
by communities of people who share an interest
in a common subject or discipline.

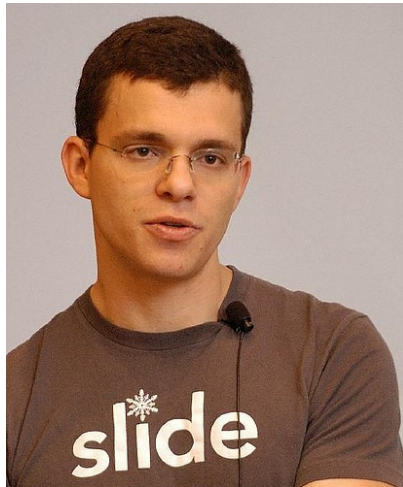
http://www.lib.uconn.edu/using/tutorials/research/HTML/Infocycle/Sources/infocycle_01.htm



ANCIENT SECRET

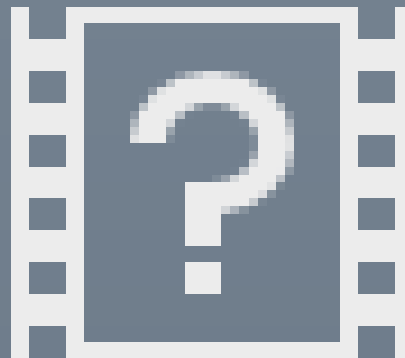
THE INVISIBLE COLLEGE

☞ Max Levchin, co-founder of PayPal



☞ Libor Michalek, co-founder of Slide.com

THE INVISIBLE COLLEGE



ANCIENT SECRET

LUCK, FATE, DESTINY

- ☿ Rod: “Competent people tend to be luckier”
- ☿ Floyd: “[Career] Luck can be highly influenced by you”
- ☿ Kohsuke Kawaguchi: Hudson Continuous integration framework, be aware of opportunities



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate() { public boolean
takeAction(FacesContext ctx, UIComponent comp) throws FacesException { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

ANCIENT SECRET

LUCK, FATE, DESTINY

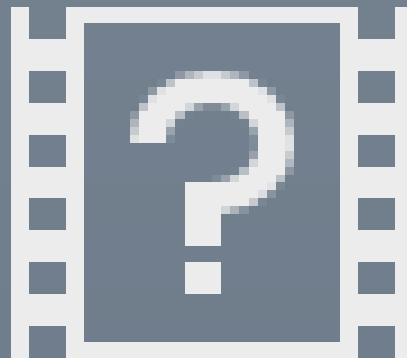
☞ Dave Thomas, Object Oriented Software
Pioneer, founder of Object Technology Inc.




```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate = false;
private FacesContext facesContext;
private UIComponent component;
private boolean isImmediate() { public boolean
takeAction() {
FacesContext facesContext;
UIComponent component;
throws FacesException;
instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

ANCIENT SECRET

LUCK, FATE, DESTINY



MODERN SECRETS

🌀 Things that have only recently emerged as useful in the work of a practicing programmer

- ★ Mastery of tools, being an optimizer and customizer
- ★ Software: how to write it, fix it, maintain it
- ★ Discerning technology trends
- ★ Motivation
- ★ Knowing when it's time to change jobs
- ★ Having a non-IT plan B

CUSTOMIZER, OPTIMIZER, MASTER OF TOOLS

🌀 The most successful programmers are often the ones who are the best with their tools.



“Smooth out the ripples in their environment”

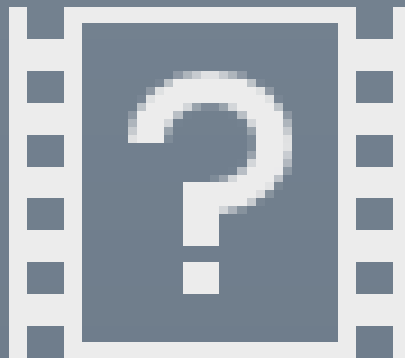
Chris Wilson, Lead Architect of Microsoft Internet Explorer



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<Component> components = new ArrayList<Component>();
public boolean takeAction(FacesContext ctx, Component comp, ActionSource as) { if (as instanceof ActionSource) {
ActionSource as = (ActionSource)as; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

MODERN SECRET

CUSTOMIZER, OPTIMIZER, MASTER OF TOOLS



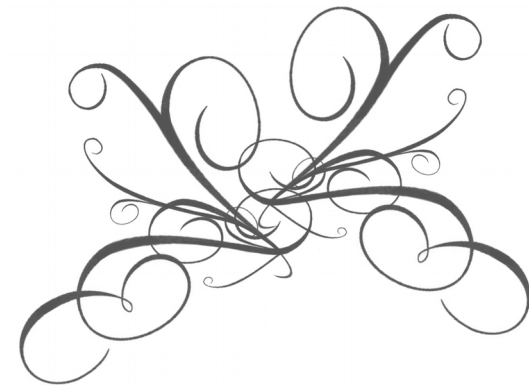

```
public class AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<Component> components = new ArrayList<Component>();
public boolean takeAction(FacesContext ctx, Component comp, ActionEvent ae) throws IOException { if (ae instanceof ActionSource) {
ActionSource as = (ActionSource)ae; if (as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

MODERN SECRET

CUSTOMIZER, OPTIMIZER, MASTER OF TOOLS



Mindset of continual optimization



MODERN SECRET

HOW TO MAKE SOFTWARE



MODERN SECRET

HOW TO MAKE SOFTWARE



- Mastery of version control
- Comprehensive Unit tests
- All software dependency management and artifact creation is automated and under version control

“It’s pretty safe to do almost anything to the code base.”



MODERN SECRET

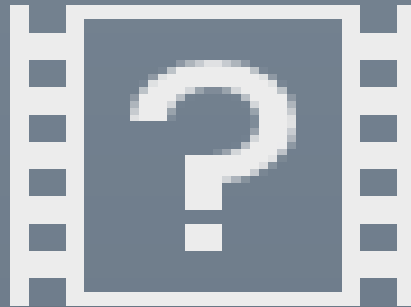
HOW TO MAKE SOFTWARE



How to write good patterns

- Ability to recognize a pattern when it comes along and bites you
- Ability to how to encapsulate the context

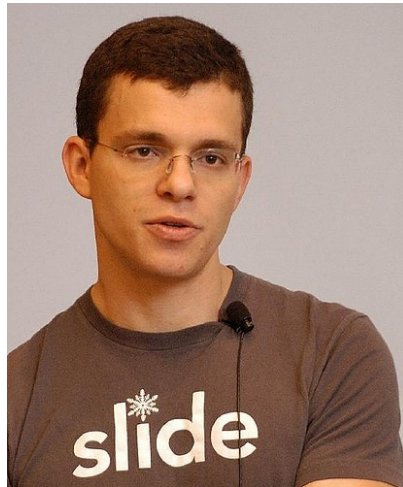
HOW TO MAKE SOFTWARE



MODERN SECRET

WHERE DOES THE PRODUCTIVITY COME FROM?

☞ Max Levchin, co-founder of PayPal



☞ Libor Michalek, co-founder of Slide.com

SPOTTING TRENDS



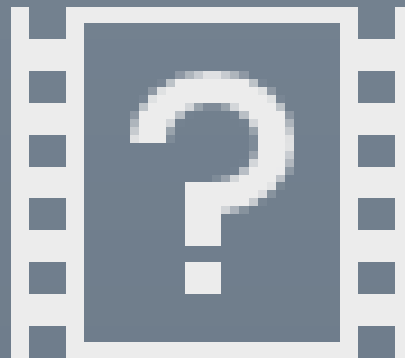
“Max: ten years”



```
public class AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private PartialTraversalViewRoot viewRoot;
private List<Component> modifiedComponents = new ArrayList<Component>();
public boolean takeAction(FacesContext ctx, UIComponent comp, ActionEvent ae) throws IOException { public boolean
takeAction(FacesContext ctx, UIComponent comp, ActionEvent ae) { if (ae instanceof ActionSource) {
ActionSource as = (ActionSource)ae; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

MODERN SECRET

SPOTTING TRENDS



SPOTTING TRENDS

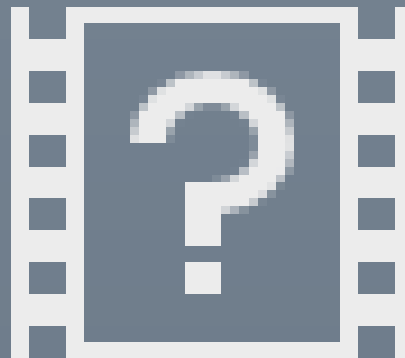


Time to get beyond Gutenberg

```
public class AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private PartialTraversalViewRoot viewRoot;
private List<Component> modifiedComponents = new ArrayList<Component>();
public boolean takeAction(FacesContext ctx, UIComponent comp, ActionEvent ae) throws IOException { public boolean
takeAction(FacesContext ctx, UIComponent comp, ActionEvent ae) { if (ae instanceof ActionSource) {
ActionSource as = (ActionSource)ae; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

MODERN SECRET

SPOTTING TRENDS




```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayDeque<Component> modifiedComponents = new ArrayDeque<>();
public boolean takeAction(FacesContext ctx, Component comp, ActionSource as) { if (as instanceof ActionSource) {
ActionSource as = (ActionSource)as; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } }
```

GETTING A NEW IDEA ADOPTED



“Getting bankers to use garbage collection”



“Find a way for the consumers of the platform to make money.”



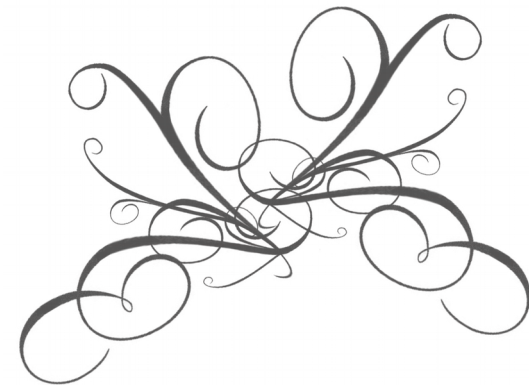
```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayDeque<Component> modifiedComponents = new ArrayDeque<>();
public boolean takeAction(FacesContext ctx, UIComponent comp, ActionSource as) { if (as instanceof ActionSource) {
ActionSource as = (ActionSource) as; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder) comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } }
```

MODERN SECRET

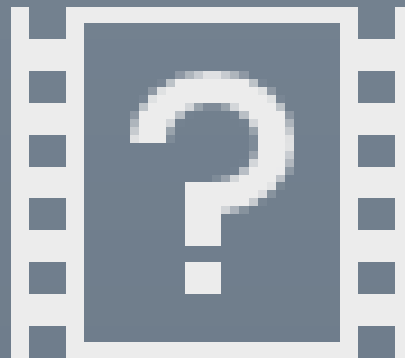
BUSINESS ACUMEN, CAREER



“How far can you progress just being a really good programmer?”



BUSINESS ACUMEN, CAREER



```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayDeque<Component> modifiedComponents = new ArrayDeque<>();
@Override public boolean takeAction(FacesContext ctx, UIComponent comp, ActionSource as) throws IOException { if (as instanceof ActionSource) {
ActionSource as = (ActionSource) as; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder) comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } }
```

MODERN SECRET

TAKE A TANGENT EVERY NOW AND THEN



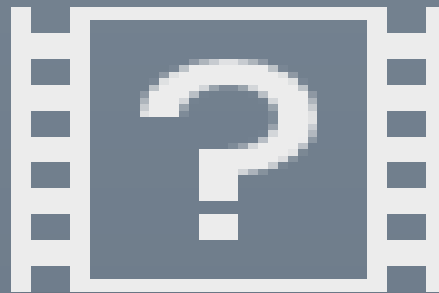
“It enabled me to stay relevant
in a very different way.”




```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private List<Component> components = new ArrayList<Component>();
public boolean takeAction(FacesContext ctx, UIComponent comp, ActionEvent evt) { public boolean
takeAction(FacesContext ctx, UIComponent comp, ActionEvent evt) { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

MODERN SECRET

TAKE A TANGENT EVERY NOW AND THEN




```
public AjaxViewRoot implements Serializable { public AjaxViewRoot(PartialTraversalViewRoot viewRoot) {if (viewRoot == null)
{throw new IllegalArgumentException("PartialTraversalViewRoot " + "cannot be null!");}setUIViewRoot(viewRoot);
}
private ArrayList<UIComponent> components = new ArrayList<UIComponent>();
private boolean isImmediate = false;
private boolean isPartialCallback() { public boolean
takeAction(FacesContext context, UIComponent comp) throws IOException { if (comp instanceof ActionSource) {
ActionSource as = (ActionSource)comp; if (!as.isImmediate()) { as.setImmediate(true); modifiedComponents.add(comp); } } else if
(comp instanceof EditableValueHolder) { EditableValueHolder ev = (EditableValueHolder)comp; if (!ev.isImmediate()) {
ev.setImmediate(true); modifiedComponents.add(comp); } } return true; } } }
```

CONCLUSION

- 🌀 Balance between humility and pride
- 🌀 Awareness of one's own ignorance
- 🌀 Questioning “Conventional Wisdom”
- 🌀 The Invisible College
- 🌀 Luck
- 🌀 Mastery of tools, being an optimizer and customizer
- 🌀 Software: how to write it...
- 🌀 Discerning technology trends
- 🌀 Motivation
- 🌀 Knowing when it's time to change jobs
- 🌀 Having a non-IT plan B

Ed Burns
<http://ridingthecrest.com/>

