

Tuning JavaServer Faces

Kito D. Mann

Principal Consultant

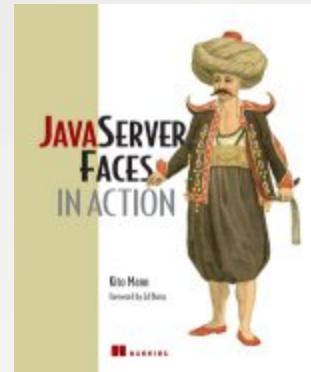
Virtua™

technical excellence.
it's that simple.

Kito D. Mann

@kito99

- » Principal Consultant at Virtua
 - » <http://www.virtua.com>
 - » Training, consulting, architecture, mentoring,
 - » Official US PrimeFaces partner
- » Author, JavaServer Faces in Action
- » Founder, JSF Central
 - » <http://www.jsfcentral.com>



Kito D. Mann

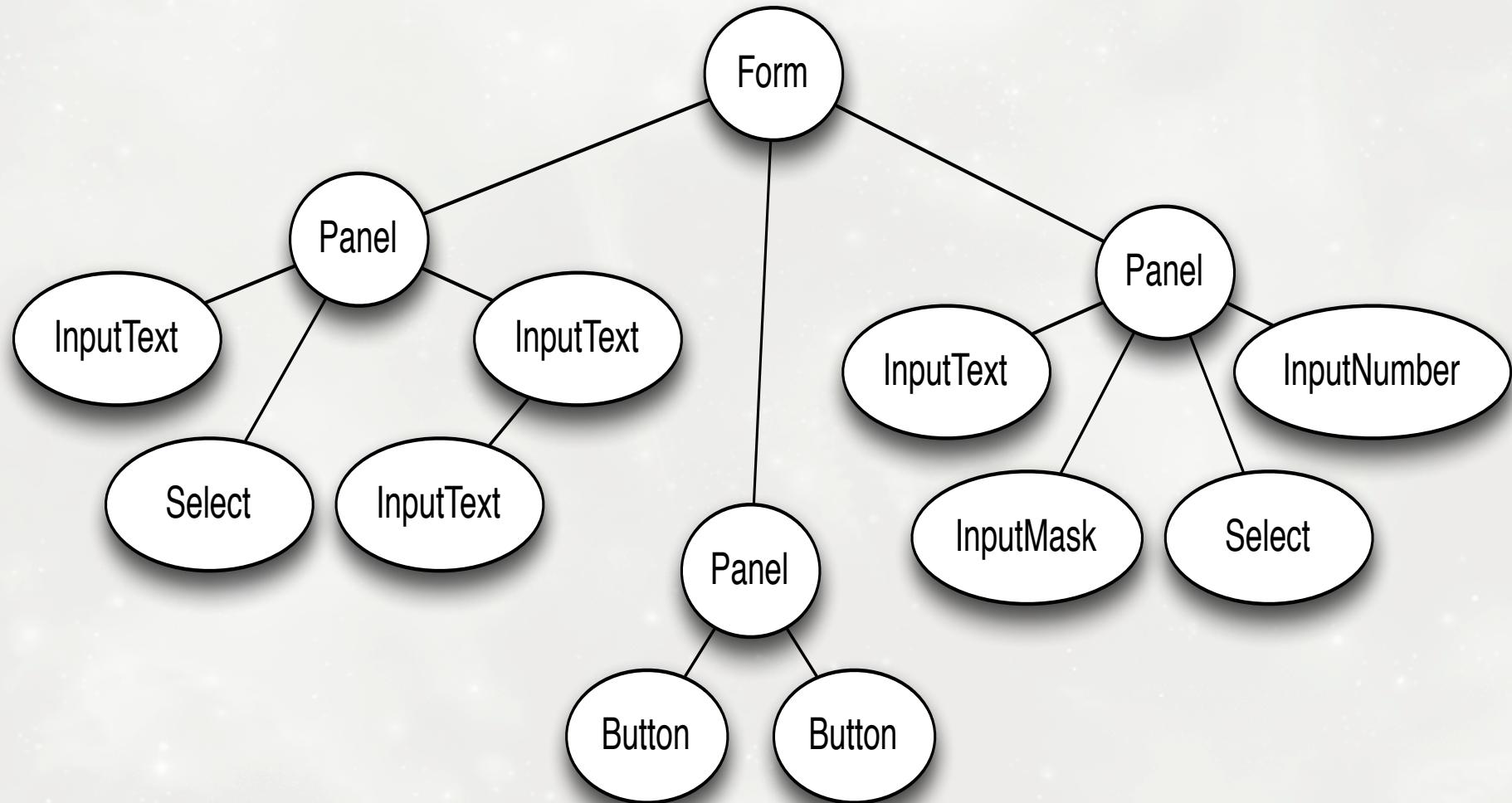
@kito99

- » Co-host, Enterprise Java Newscast
 - » <http://enterprisejavane.ws.com>
- » Internationally recognized speaker
 - » JavaOne, JavaZone, Devoxx, NFJS, TSSJS, etc.
- » JCP Member
 - » JSF, MVC, CDI, JSF Portlet Bridge, Portlets

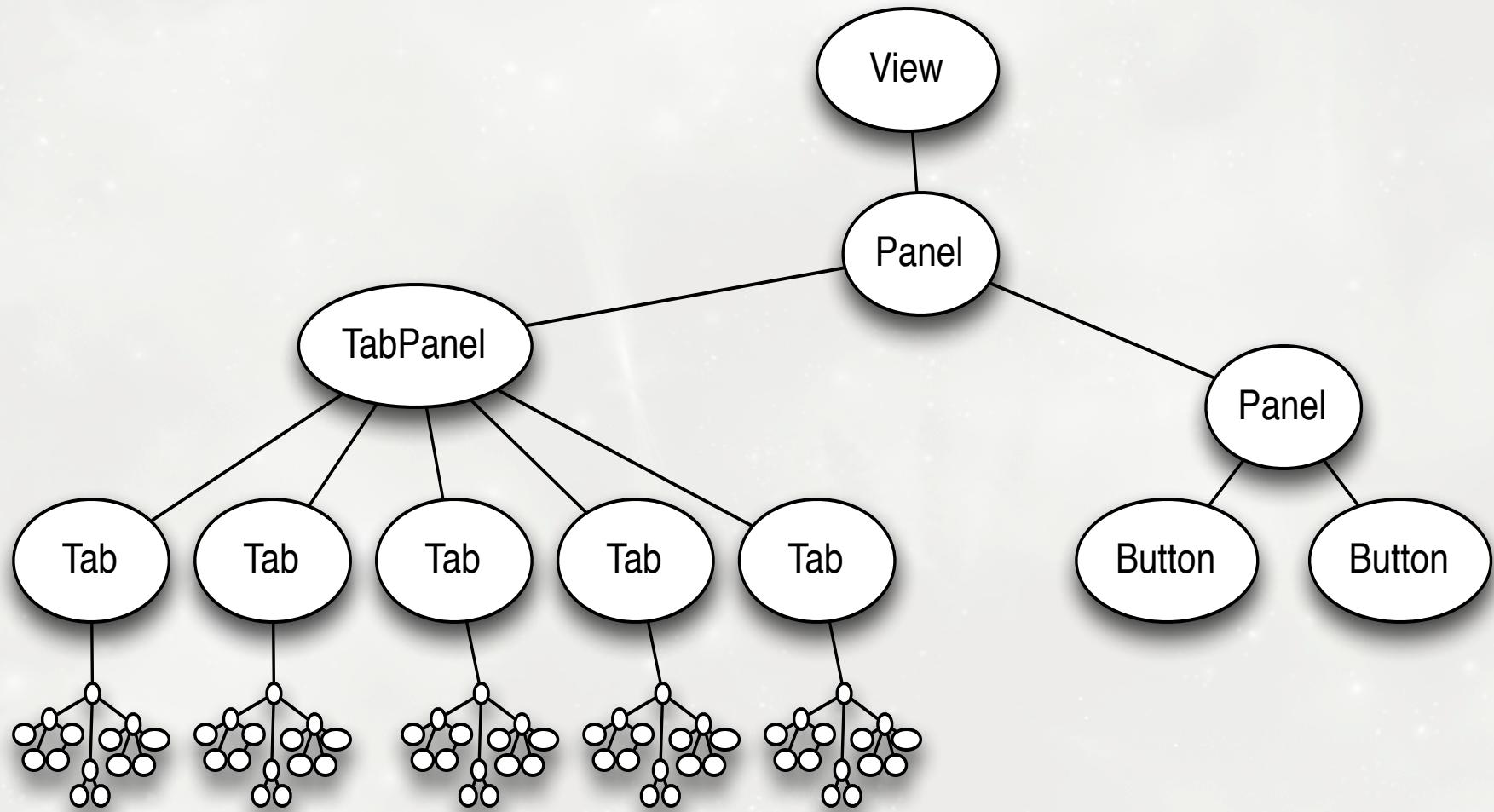
Questions?

What impacts performance?

- » Page size
- » Session state
 - » Component tree
 - » Application state
- » Request size
- » EL Evaluation
- » Configuration Flags
- » Accessing FacesContext



Bloated component tree



Bloated component tree

- » Consume lots of memory
- » Render large pages
- » Slower

Bloated component tree

- » The rendered attribute does **not** affect the size of the component tree

Bloated component tree

- » Is Ajax really required?
- » Consider transient markup (<div>, , text output, etc.)
- » Consider stateless views (JSF 2.2)
- » Investigate dynamic loading features

Bloated component tree

- » Use a scope other than session
- » Use <c:if> (sparingly)
- » Split into separate views

Session overload

Session overload

- » Put beans in request scope when possible
- » For Ajax, use view scope

Session overload

- » Flow Scope
 - » JSF 2.2
- » Conversation scope
 - » JSF 2: CDI

```
public void getData() {  
    return service.getData();  
}
```

Bloated Requests

Use PartialSubmit

False

Post

Data:javax.faces.partial.ajax=true&javax.faces.source=form
%3Aj_idt18&javax.faces.partial.execute=form
%3Aname&javax.faces.partial.render=form
%3Adisplay&form%3Aj_idt18=form
%3Aj_idt18&form=form&form%3Aname=&form
%3Aj_idt20=&form%3Aj_idt22=&form%3Aj_idt24=&form
%3Aj_idt26=&form%3Aj_idt28=&form%3Aj_idt30=&form
%3Aj_idt32=&form%3Aj_idt34=&form%3Aj_idt36=&form
%3Aj_idt38=&form%3Aj_idt40=&form%3Aj_idt42=&form
%3Aj_idt44=&form%3Aj_idt46=&form%3Aj_idt48=&form
%3Aj_idt50=&form%3Aj_idt52=&form%3Aj_idt66=&form
%3Aj_idt68=&form%3Aj_idt70=&form%3Aj_idt72=&form
%3Aj_idt74=&form%3Aj_idt76=&form%3Aj_idt78=&form
%3Aj_idt80=&form%3Aj_idt82=&form%3Aj_idt84=&form
%3Aj_idt86=&form%3Aj_idt88=&form%3Aj_idt90=&form
%3Aj_idt92=&form%3Aj_idt94=&form%3Aj_idt96=&form
%3Aj_idt98=&javax.faces.ViewState=791648252190942198
3%3A-2156051337299048496

True

Post

Data:javax.faces.partial.ajax=true&javax.faces.source=form
%3Aj_idt18&javax.faces.partial.execute=form
%3Aname&javax.faces.partial.render=form%3Adisplay&form
%3Aj_idt18=form%3Aj_idt18&form=form&form
%3Aname=&javax.faces.ViewState=7916482521909421983
%3A-2156051337299048496

Getter grabber

```
public void getData() {  
    return service.getData();  
}
```

Getter grabber

- » Do not perform any expensive operations in getters
- » Retrieve data when view is loaded

Implementation Configuration Flags

Unnecessary FacesContext lookups

- » Calling FacesContext.getCurrentInstance()
multiple times in the same method

Resources

- » [http://www.jsfcentral.com/articles/
understanding jsf performance 1.html](http://www.jsfcentral.com/articles/understanding_jsf_performance_1.html)
- » [http://www.jsfcentral.com/articles/
understanding jsf performance 2.html](http://www.jsfcentral.com/articles/understanding_jsf_performance_2.html)
- » [http://www.jsfcentral.com/articles/
understanding jsf performance 3.html](http://www.jsfcentral.com/articles/understanding_jsf_performance_3.html)
- » [http://www.jsfcentral.com/articles/
speed up your jsf app 1.html](http://www.jsfcentral.com/articles/speed_up_your_jsf_app_1.html)
- » [http://www.jsfcentral.com/articles/
speed_up_your_jsf_app_2.html](http://www.jsfcentral.com/articles/speed_up_your_jsf_app_2.html)