imagine 2012

Magento® Conference

Magento 2
View Layer Evolution

Sergey Shymko

Software Engineer, Magento,

a division of X.commerce, Inc. (an eBay company)

Introduction

- Theme/Skin – graphical appearance of an application
- Magento distinguishes theme & skin
  - Responsibility
  - File types
  - Web-accessibility
- Design package (instance) – group of themes/skins

Magento Theme

- Theme – markup & text, blocks presence & positioning
- Theme files – dynamically processed, no web access

Magento Skin

- Skin – look and feel of a theme, colors & styles
- Multiple skins may exist for a theme
- Skin files – static, web-accessible
  - Images
  - CSS
  - JavaScript
  - etc.

Magento 1.x Skin

- Skins are located separately from themes
- Any skin can be chosen for any theme
- **Incompatible skin can be chosen for a theme**

"example" package

"default" theme ← "simple" skin

"default" skin

"modern" skin → "mobile" theme
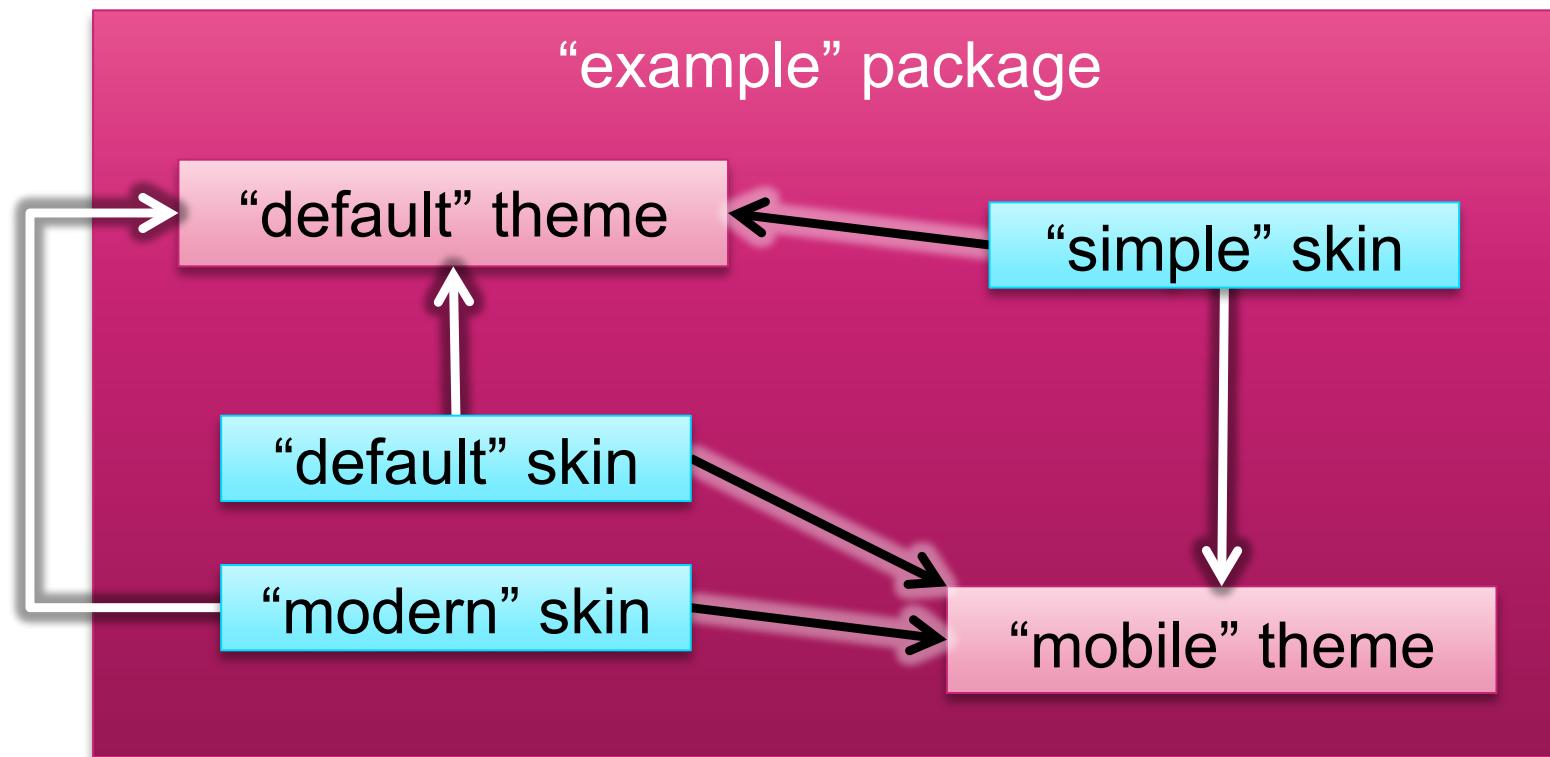
Magento 2 Skin

- Skin is located within a theme
- Skin belongs to one theme only
- **Skin files web-access challenge**
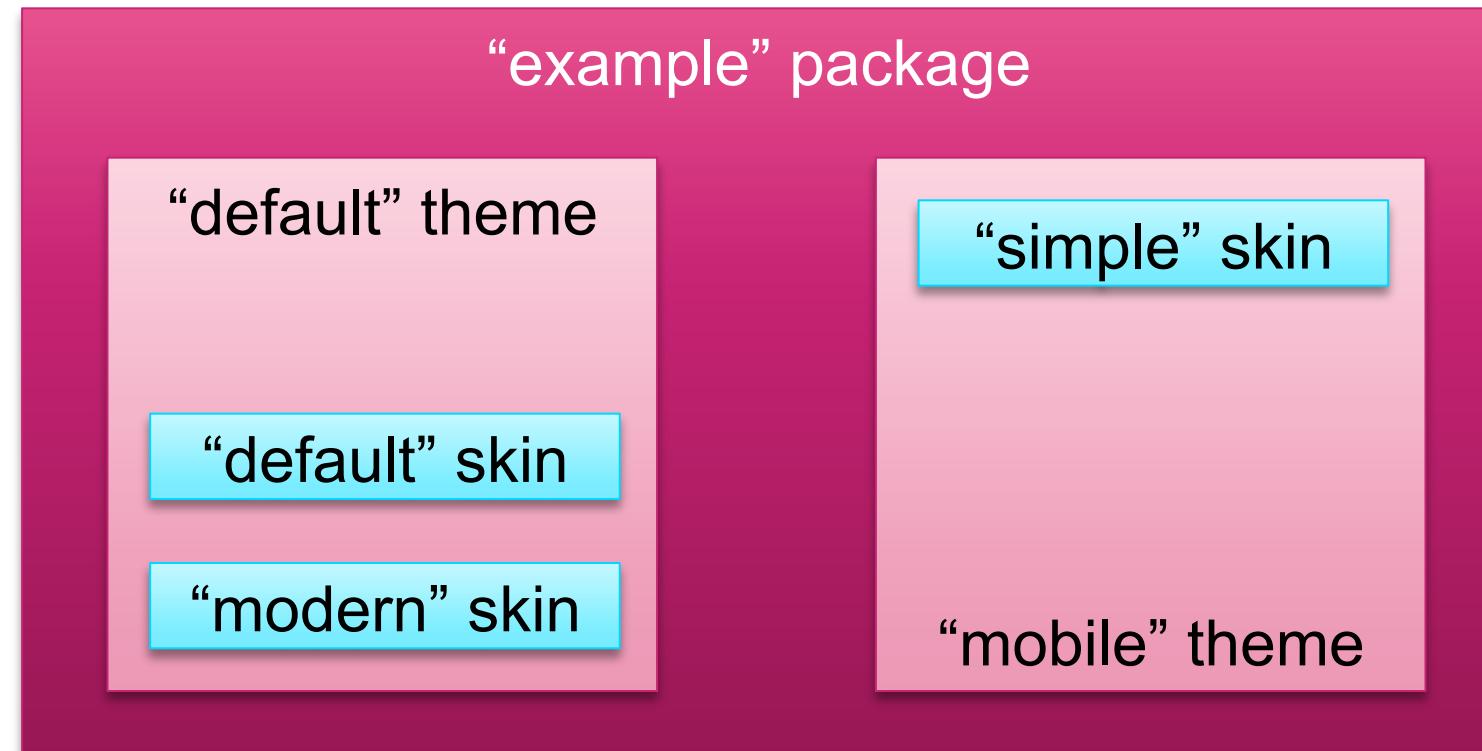
Magento 1.x Skin Localization

- Text in images, JavaScript, etc. needs localization
- New skin for every locale
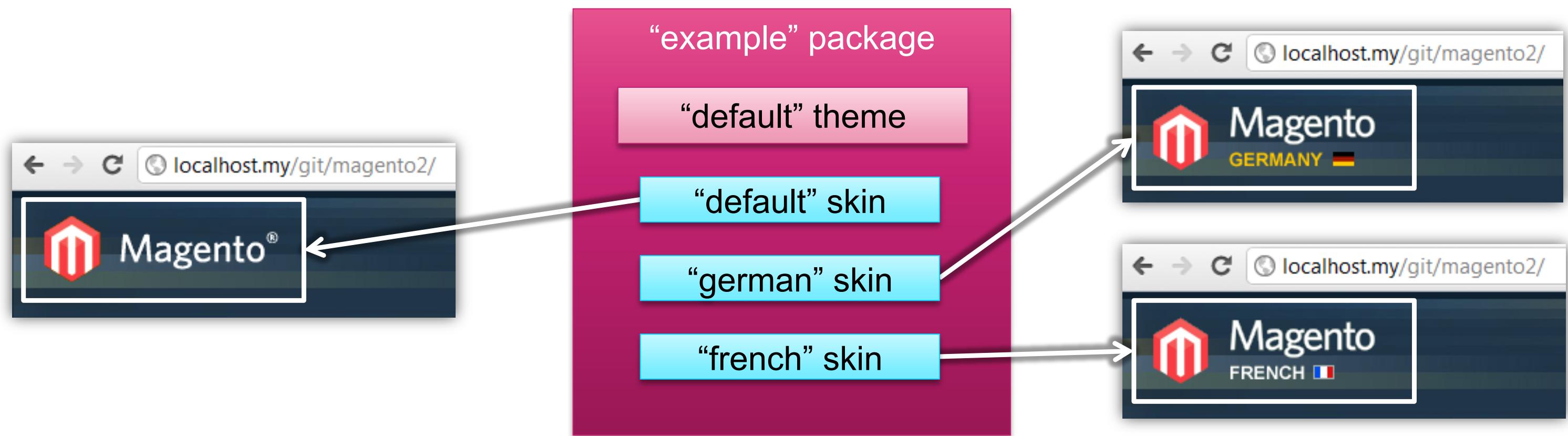
Magento 2 Skin Localization

- Skin provides files for multiple locales
- Better skin files reuse
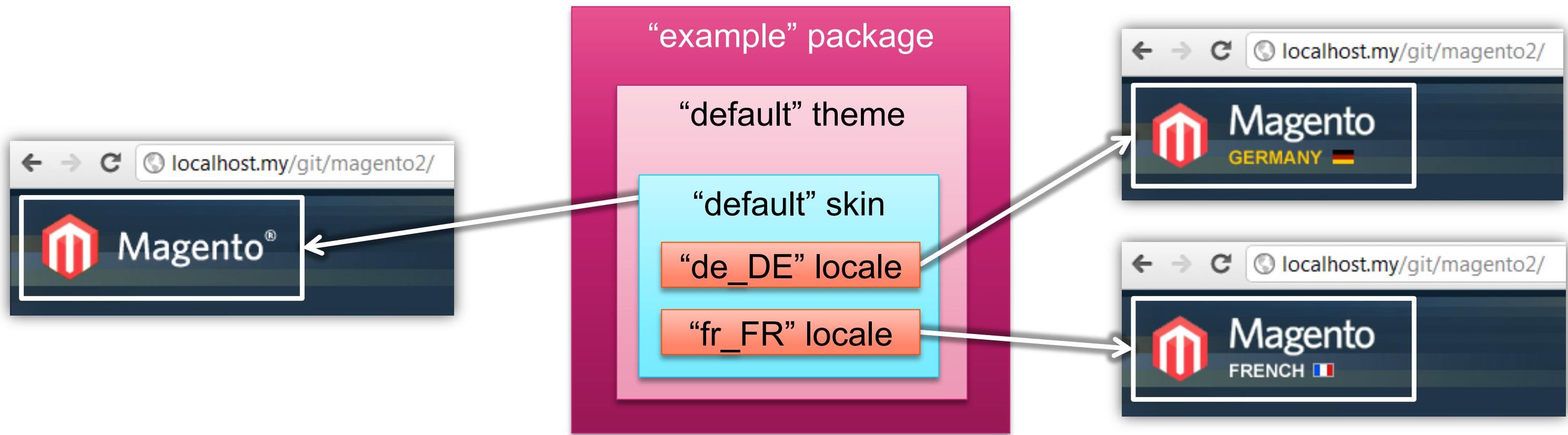
Magento 2

View Layer Modularity

Magento 1.x View Layer Fallback

	Package	Theme/Skin
1.	<current>	<current>
2.	<current>	<default>
3.	<current>	"default"
4.	"base"	"default"

<xxx> – configured name

"xxx" – exact name

imagine 2012
Magento Conference
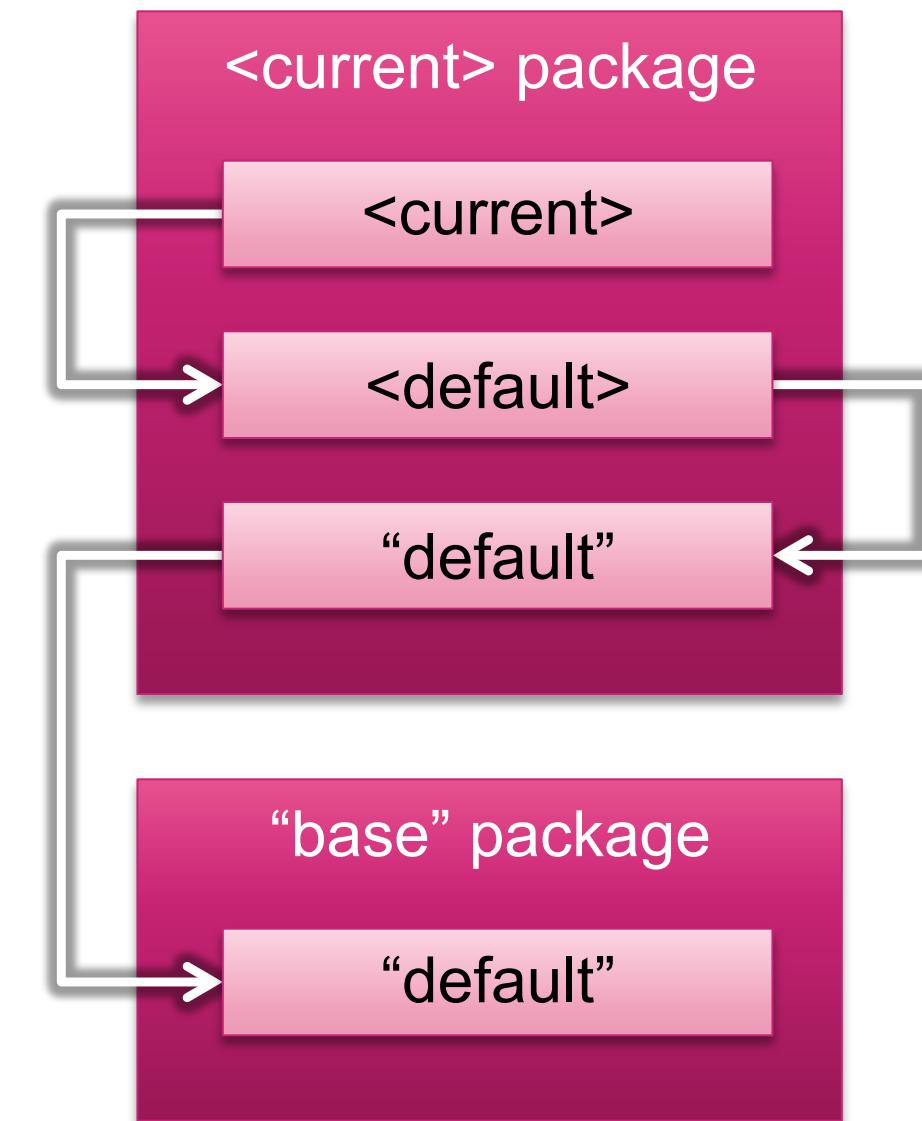
Magento 1.x "base/default" Theme/Skin

- Contains pseudo-modular view files

  📁 `app/design/frontend/base/default/…`
  - 📁 `catalog`
  - 📁 `checkout`
  - 📁 `cms`   } module aliases
  - 📁 `sales`
  - 📁 `…`
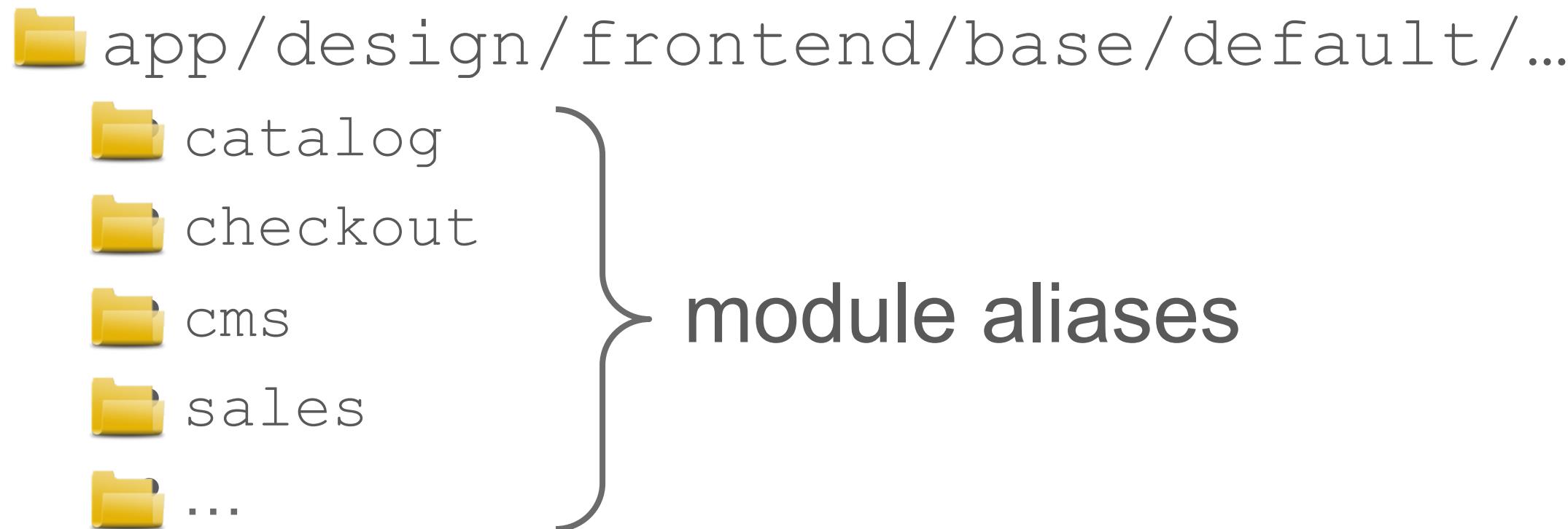
Modular View Files

- Module view files
  📁 `app/code/<pool>/<namespace>/<module>/view`
- View files are divided by areas
- Theme & skin files reside together
- Theme/skin files distinguished by extension
- Theme/skin can refer to module's files
- Module cannot refer to theme/skin

Module Scope

- Referring modular file

  `<Namespace_Module>::file.ext`

- Omit module scope

  - If template used solely within its own module

  - *In module's configuration file**

  - *In module's layout update file**

  - *In module's CSS file**

* - module detection is possible, not implemented yet

Theme Files Fallback

Package	Theme
1. <current>	<current>
2. ~~<current>~~	~~<default>~~
2. <current>	"default"
3. ~~"base"~~	~~"default"~~
3. Module theme files	

<current> package

  <current> theme

  "default" theme

Module files

<xxx> – configured name

"xxx" – exact name

imagine 2012
Magento Conference
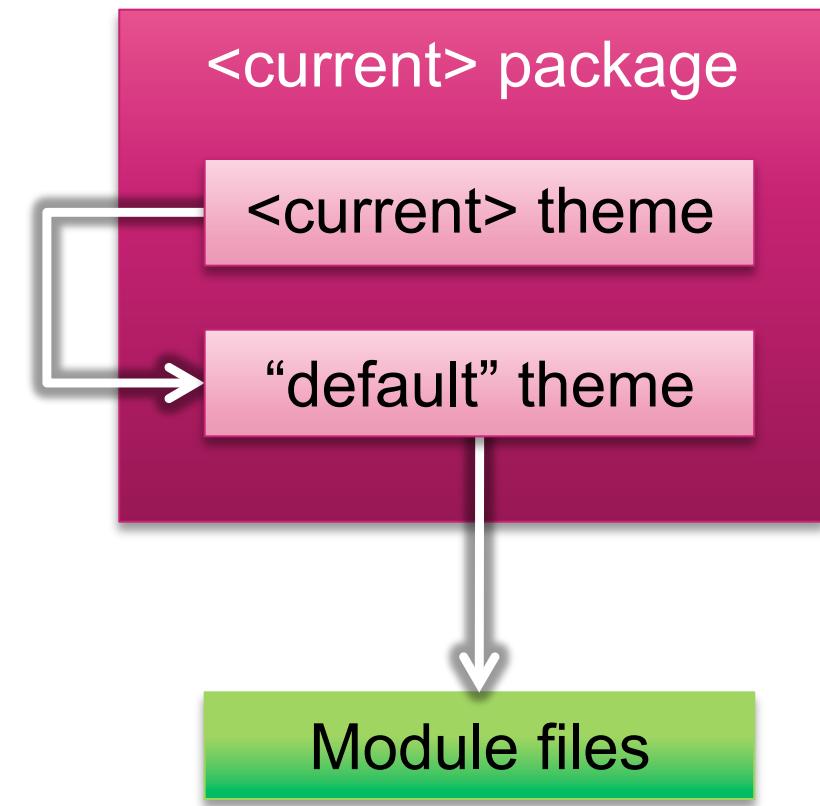
Skin Files Fallback

Theme	**Skin**
1. <current>	<current>
2. <current>	"default"
3. "default"	<current>
4. "default"	"default"

5. Module skin files

6. JavaScript libraries
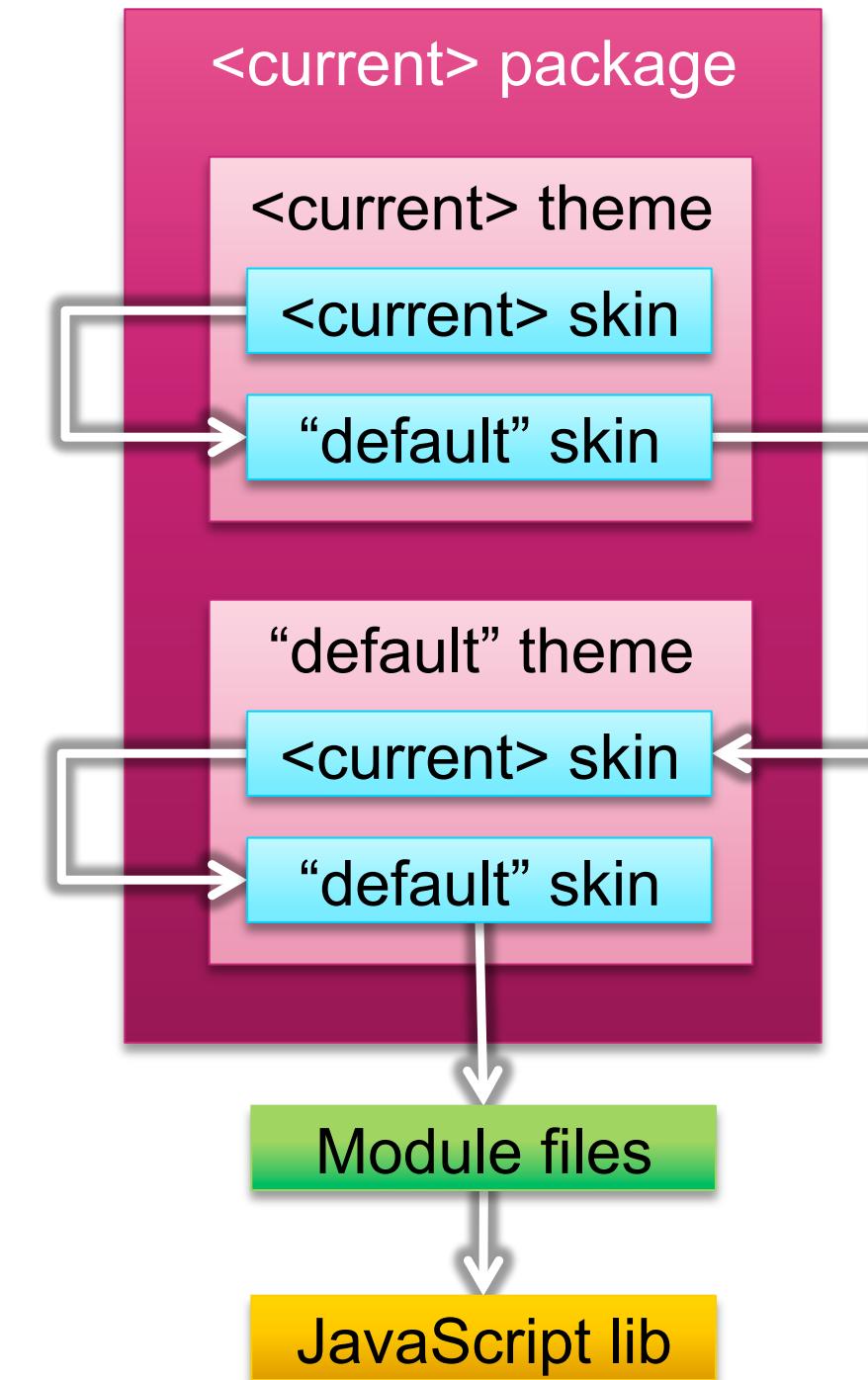
JavaScript Libraries Fallback

- Single interface for skin and JS lib files
  - Simplified HTML `<head>` tag manipulation
- Skin can override JS lib files
  - Skin may require specific version of JS lib
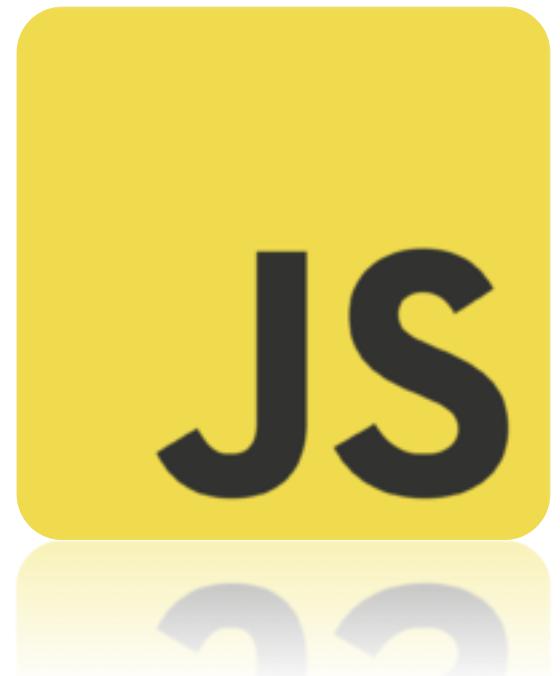
Skin Files Publication

- Publication – copying to web-accessible location
  - 📁 `pub/skin`
- Need publication
  - 📁 `app/code/<pool>/<namespace>/<module>/view`
  - 📁 `app/design/<area>/<package>/<theme>/skin/<skin>`

CSS Files Processing

- Need processing
  - Relative URLs
  - URLs with module scope

- Don't need processing
  - Absolute URLs

Magento 2

Page Types

Magento 1.x Page Rendering

- Layout handle – group of instructions which define visual elements positioning & properties

- Controller uses arbitrary layout handles to render a page

- Used layout handle not always corresponds to action name

- Any combination of layout handles can be used
  - No way to determine actual number of pages a website has

Page Types

- Page type – declaration of visual elements & their properties
- Page – page type rendered with the actual data
- Page type may inherit another page type
- Controller uses one page type only
- Inherited page types are loaded implicitly

Magento 2

Containers

Container vs Block

- Block – unit of a page output
- Container – unit of a page structure
- Container consists of blocks and other containers
- Container renders all children
- Converted to containers
  - `Mage_Core_Block_Text_List`
  - `Mage_Page_Block_Html_Wrapper`

Containers in Layout Updates

`<container>` directive attributes:

- `name` – unique name
- `label` – human-readable name

```
<layout_handle>
    <container name="root_container" label="Root Container">
        <block type="Mage_Core_Block_Template" name="block1" template="1.phtml"/>
        <container name="nested_container" label="Nested Container"/>
    </container>
```

Containers & Page Types Usage

- Visual Design Editor
  - Page type breadcrumbs & navigation
  - Drag-and-drop of blocks & containers

Summary

- Magento 2 Features
  - Skin files localization
  - Modular view files
  - Page types
  - Containers
- Links
  - github.com/magento/magento2
  - wiki.magento.com/display/MAGE2DOC
  - wiki.magento.com/display/MAGE2PROJECT

imagine 2012
Magento Conference

Thank You!
Q & A

Sergey Shymko

sergey.shymko@magento.com

Thank You!
Questions & Answers

Sergey Shymko
sergey.shymko@magento.com