

M2 CPFED

1. Themes /

Design Config

John Hughes

Head of Magento - Fisheye Media Ltd.

Overview.

- Components, areas and scopes
- Theme location and structure
- Theme folders and files
- Theme hierarchy and inheritance
- Theme configuration
- Theme scheduling

Get ready, this is the first of many!



Components areas and scopes

Components.

- **What are the different component types?**
- What are their purposes?

Components.

What are the different component types?

- **Modules**
- **Themes**
- **Language packages**

Components.

- What are the different component types?
- **What are their purposes?**

Components.

Type	Purpose
Modules	Features / functionality 'Business logic'
Themes	Appearance
Language packages	Internationalisation / Translations

Areas.

- **What are the different areas in Magento 2?**
- What are their purposes?
- Which are relevant to themes?

Areas.

What are the different areas in Magento 2?

- **frontend**
- **adminhtml**
- **base**
- **crontab**
- **webapi_rest**
- **webapi soap**

Areas.

- What are the different areas in Magento 2?
- **What are their purposes?**
- Which are relevant to themes?

Areas.

Area	Usage
frontend	Storefront only
adminhtml	Admin panel only
base	Storefront + admin panel
crontab	Scheduled tasks
webapi_rest	REST API
webapi soap	SOAP API

Areas.

- What are the different areas in Magento 2?
- What are their purposes?
- **Which are relevant to themes?**

Areas.

Which are relevant to themes?

- **frontend**
- **adminhtml**
- **base**

Scopes.

- **What are scopes?**
- What are the different scopes in Magento 2?
- What are their purposes?
- How do scopes relate to one another?

Scopes.

What are scopes?

- Magento supports highly flexible ‘multi-store’ functionality
- This allows variations to language, design, currency, product catalogue, config etc. from one ‘Magento instance’
- You can even have multiple completely discrete websites running from a single ‘Magento instance’

*One Magento instance = one install / codebase

Scopes.

- What are scopes?
- **What are the different scopes in Magento 2?**
- What are their purposes?
- How do scopes relate to one another?

Scopes.

What are the different scopes in Magento 2?

- **Global (default)**
- **Website**
- **Store (Group)**
- **Store View**

Scopes.

- What are scopes?
- What are the different scopes in Magento 2?
- **What are their purposes?**
- How do scopes relate to one another?

Scopes.

Scope Type	General usage
Global (default)*	Applicable to entire Magento instance (default settings)
Website*	For discrete brands, payment currencies, shipping, products
Store (Group)	Change category structure
Store View*	Theme, design, locale, display currency + other config

*themes can be managed at these scopes

Scopes.

- What are scopes?
- What are the different scopes in Magento 2?
- What are their purposes?
- **How do scopes relate to one another?**

Scopes.

How do scopes relate to one another?

- Scope types are hierarchical and fallback to / inherit from the parent scope

Scope fallback.

Store view



Store (group)



Website



Global (default)

Theme location and structure

Theme location.

- **What are 2 ways of installing a theme?**
- What are the locations of these?
- What are the use cases for using each?

Theme location.

What are 2 ways of installing a theme?

- Direct
- Composer

Theme location.

- What are 2 ways of installing a theme?
- **What are the locations of these?**
- What are the use cases for using each?

Theme location.

Method	Location(s)
Direct	<code>app/design/<area></code> <code>app/design/frontend</code> <code>app/design/adminhtml</code>
Composer	Technically anywhere Normally in <code>vendor/</code>

Theme location.

- What are 2 ways of installing a theme?
- What are the locations of these?
- **What are the use cases for using each?**

Theme location.

Method	Use case
Direct	Theme(s) specific to the project / merchant
Composer	Reusable / base theme 3rd party theme

Theme structure.

- **What 3 elements define a theme?**

Theme structure.

Area	e.g. frontend
Package name	e.g. FisheyeAcademy
Theme name	e.g. training

app/design/**frontend/FisheyeAcademy/training**

Default themes.

- **What are the default themes in Magento 2?**

Default themes.

Name	Area	Location
Blank	frontend	vendor/magento/theme-frontend-blank
Luma*	frontend	vendor/magento/theme-frontend-luma
Backend	adminhtml	vendor/magento/theme-adminhtml-backend

*Luma is a child theme of Blank

Theme folders and files

Key theme files.

- **What files are required and why?**

Key theme files

File	Purpose
registration.php*	Registers theme with Magento
theme.xml*	Theme name Parent theme association
composer.json	To enable installation via composer

*required files

registration.php

<?php

```
\Magento\Framework\Component\ComponentRegistrar::  register (  
    \Magento\Framework\Component\ComponentRegistrar::  THEME ,  
    'frontend/FisheyeAcademy/training' ,  
    __DIR__  
);
```

registration.php

```
<?php
```

```
\Magento\Framework\Component\ComponentRegistrar::register(  
    \Magento\Framework\Component\ComponentRegistrar:: THEME ,  
    'frontend/FisheyeAcademy/training',  
    __DIR__  
);
```

Component type



registration.php

```
<?php
```

```
\Magento\Framework\Component\ComponentRegistrar::register(  
    \Magento\Framework\Component\ComponentRegistrar::THEME,  
    'frontend/FisheyeAcademy/training' ,  
    __DIR__  
);
```

Location



theme.xml

```
<?xml version="1.0"?>

<theme xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="urn:magento:framework:Config/etc/theme.xsd">

    <title> Fisheye Academy Training Theme </title>

    <parent> Magento/luma </parent>

</theme>
```

theme.xml

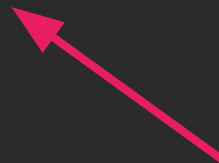
```
<?xml version="1.0"?>
<theme xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="urn:magento:framework:Config/etc/theme.xsd">
    <title> Fisheye Academy Training Theme </title>
    <parent>Magento/luma</parent>
</theme>
```

Theme name



theme.xml

```
<?xml version="1.0"?>
<theme xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="urn:magento:framework:Config/etc/theme.xsd">
    <title>Fisheye Academy Training Theme</title>
    <parent>Magento/luma </parent>
</theme>
```



Parent theme

Theme folders / files.

- **What other folders / files are found in themes?**
- **What are their purpose?**

Theme folders / files.

Folder / File(s)	Purpose
etc/view.xml	Theme configuration values e.g. image sizes, product gallery settings, JS bundling file / folder exclusions
i18n/<locale_COUNTRY>.csv e.g i18n/en_GB.csv	Translations

Theme folders / files.

Folder / File(s)	Purpose
media/preview.jpg	Preview image of theme For admin panel Image filename can be changed in theme.xml
web/css/	Theme specific public CSS / LESS files
web/css/source	Theme specific LESS component files

Theme folders / files.

Folder / File(s)	Purpose
web/fonts	Theme specific web fonts
web/images	Theme specific images Not CMS images (e.g. banners)
web/js	Theme specific javascript

Theme folders / files.

Folder / File(s)	Purpose
<Vendor>_<Module>/ e.g. Magento_Cms/ Magento_Catalog/ Magento_Theme/ etc.	Module file overrides: <ul style="list-style-type: none">● Templates<ul style="list-style-type: none">○ PHP (.phtml)○ Knockout (.html)○ Email (.html)● Layout XML*● LESS / CSS● Images● Javascript

*layout XML is merged by default, not overridden

Theme hierarchy and inheritance

Theme hierarchy.

- **What are child themes for?**
- How are parent / child relationships managed?
- What is the difference between a parent and a child theme?

Theme hierarchy.

What are child themes for?

- To extend an existing theme

Theme hierarchy.

- What are child themes for?
- **How are parent / child relationships managed?**
- What is the difference between a parent and a child theme?

Theme hierarchy.

How are parent / child relationships managed?

- In the `<parent/>` node in `theme.xml`

Theme hierarchy.

- What are child themes for?
- How are parent / child relationships managed?
- **What is the difference between a parent and a child theme?**

Theme hierarchy.

What is the difference between a parent and a child theme?

- **No difference**
 - Parent themes can still be used

Theme inheritance.

- **How are files overridden?**
- How are module specific files overridden?
- What is the inheritance / fallback process?

Theme inheritance.

How are files overridden?

- By placing the relevant file in the **same location** in the child theme

Theme inheritance.

Parent theme location

- `vendor/magento/theme-frontend-luma/etc/view.xml`

Child theme location

- `app/design/frontend/FisheyeAcademy/training/etc/view.xml`

Theme inheritance.

- How are files overridden?
- **How are module specific files overridden?**
- What is the inheritance / fallback process?

Theme inheritance.

How are module specific files overridden?

- By placing the relevant file in the specified location within the **module subdirectory** of the theme

Theme inheritance.

Module location (relative)

- **view/**<area>/path/to/file.type

Theme location (relative)

- <Vendor_Module>/path/to/file.type

Theme inheritance.

Module location (relative) - **Magento_Theme**

- **view/frontend/templates/html/logo.phtml**

Theme location (relative)*

- **<Magento_Theme>/templates/html/logo.phtml**

*Theme must be in the 'frontend' area

Theme inheritance.

- How are files overridden?
- How are module specific files overridden?
- **What is the inheritance / fallback process?**

Theme inheritance.

Child theme



Parent theme(s)



Module frontend/adminhtml area



Module base area

Theme inheritance.

app/design/frontend/FisheyeAcademy/training/Magento_Catalog
/templates/product/price/final_price.phtml



vendor/magento/theme-frontend-luma/Magento_Catalog
/templates/product/price/final_price.phtml



vendor/magento/theme-frontend-blank/Magento_Catalog
/templates/product/price/final_price.phtml



Theme inheritance.



vendor/magento/module-catalog/view/frontend
/templates/product/price/final_price.phtml



vendor/magento/module-catalog/view/base
/templates/product/price/final_price.phtml

Theme configuration

Theme configuration.

- **Where can a list of themes be found?**
- Where are themes configured?
- How are scopes involved?
- What can be configured?

Theme configuration.

Where can a list of themes be found?

- Installed themes are listed in the admin panel:

CONTENT > Design > Themes

Theme configuration.

- Where can a list of themes be found?
- **Where are themes configured?**
- How are scopes involved?
- What can be configured?

Theme configuration.

Where are themes configured?

- Themes are configured in the admin panel:

CONTENT > Design > Configuration

Theme configuration.

- Where can a list of themes be found?
- Where are themes configured?
- **How are scopes involved?**
- What can be configured?

Theme configuration.

How are scopes involved?

- Themes can be set / design configured:
 - **Globally**
 - **Per website**
 - **Per store view**

Theme configuration.

- Where can a list of themes be found?
- Where are themes configured?
- How are scopes involved?
- **What can be configured?**

Theme configuration.

What can be configured?

- Design configuration includes:
 - **HTML <head> info (meta data etc.)**
 - **Header / footer config (logo etc.)**
 - **Misc. scripts inclusion**
 - **Image watermarks**
 - **Pagination**
 - **Email config (logo etc.)**

Further theme considerations.

- **What happens if theme config is changed?**
- What happens if a theme is added?
- What happens if a theme is removed?
- What are common mistakes to be aware of?

Further theme considerations.

What happens if theme config is changed?

- The values are updated in the ``core_config_data`` table* for the relevant scope
- **Configuration** and **(Full) Page** cache need refreshing in order for changes to be reflected on the frontend

*This is the same table as all other store config is stored in

Further theme considerations.

- What happens if theme config is changed?
- **What happens if a theme is added?**
- What happens if a theme is removed?
- What are common mistakes to be aware of?

Further theme considerations.

What happens if a theme is added?

- New themes will be added to ``theme`` table
- They are then available to assign to a scope within the configuration pages

Further theme considerations.

- What happens if theme config is changed?
- What happens if a theme is added?
- **What happens if a theme is removed?**
- What are common mistakes to be aware of?

Further theme considerations.

What happens if a theme is removed?

- Removing a themes files does not remove it from the **`theme`** table
- If the theme was assigned at any scope, the scope will fallback to the 'default theme' (**Magento/luma**)

Further theme considerations.

- What happens if theme config is changed?
- What happens if a theme is added?
- What happens if a theme is removed?
- **What are common mistakes to be aware of?**

Further theme considerations.

What are common mistakes to be aware of?

- Assuming config values are stored in a separate database table to ``core_config_data``
- Assuming config is linked to themes rather than scopes
- Setting the wrong values at the wrong scope or not being aware values are overwritten at a another scope

Theme scheduling

Theme scheduling.

- **What is theme scheduling?**
- How can schedules be managed?
- How does it work?
- How does caching impact scheduled changes?

Theme scheduling.

What is theme scheduling*?

- The ability to **override** the theme for a given store view for a specified period of time
- Fully **automated** i.e. can be set up in advance

*Not to be confused with the infinitely more powerful 'staging and preview' feature in Magento Commerce

Theme scheduling.

- What is theme scheduling?
- **How can schedules be managed?**
- How does it work?
- How does caching impact scheduled changes?

Theme scheduling.

How can schedules be managed?

- Themes overrides are scheduled in the admin panel:

CONTENT > Design > Schedule

Theme scheduling.

- What is theme scheduling?
- How can schedules be managed?
- **How does it work?**
- How does caching impact scheduled changes?

Theme scheduling.

How does it work?

- Scheduled theme configuration requires:
 - **The store view to apply to** (other scopes not supported)
 - **The theme to change to**
 - **Start and end date**

Theme scheduling.

How does it work?

- **Only one** scheduled theme change can be assigned to the **same store** within the **same time frame**
 - i.e. you can't overlap dates
 - Magento will error if trying to create / save a scheduled theme change with overlapping dates

Theme scheduling.

- What is theme scheduling?
- How can schedules be managed?
- How does it work?
- **How does caching impact scheduled changes?**

Theme scheduling.

How does caching impact scheduled changes?

- Full Page Cache (FPC) **overrides** scheduled changes
- Magento **does not** purge FPC when a scheduled theme starts or ends
- However, Magento does purge cache via a scheduled cron task at **2.30am each day**

Wrap up / resources

Code examples.

<https://github.com/fisheye-academy/theme-frontend-training>

Code challenge.

- Create your own frontend theme in **app/design**
- Make some modifications & check they display
 - e.g. **etc/view.xml** config
 - Set / change parent themes in **theme.xml**
- Assign your theme to the default storefront

Bonus challenge.

- Find out why sample data CMS content has disappeared when your theme is active

Credit / Resources / Thanks.

- **Official Magento Study Guide**
- **Swift Otter Study Guide**
- **Magento DevDocs**