

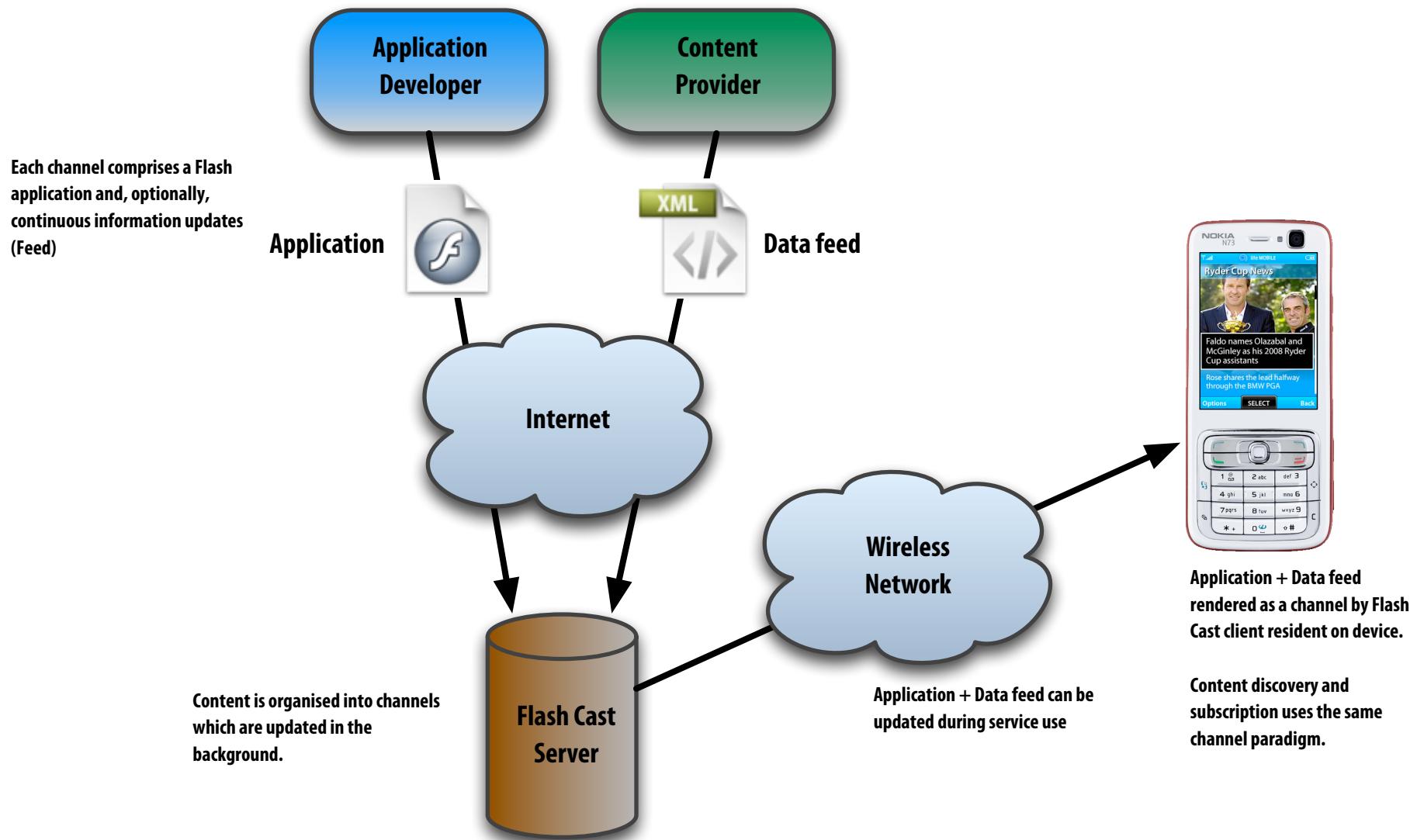
# Flash Cast - A guide for content developers

For Flash Cast v1.2

# Flash Cast Overview

- **What is Flash cast?**
- **What is a channel?**

Flash Cast is a client server platform for delivering rich data services to handsets



### A Channel normally comprises a data feed and an application

```
<data>
<type name="Ryder Cup News">
<item id="news01">
<prop name="headline">
  Faldo names Olazabal and McGinley as his 2008 Ryder Cup
  assistants
</prop>
<prop name="image" url="http://www.rydercupgolf.com/images/
faldo.jpg"/>
<prop name="body">
  European Ryder Cup captain Nick Faldo has named Jose
  Maria Olazabal and Paul McGinley as his two assistants for
  next year's competition in America. But Faldo, 49, said he
  would select late alternatives should either of them qualify
  to play again. Spain's Olazabal, 41, has played in seven
  Ryder Cups, while Ireland's McGinley, 40, has won three
  caps. "These are two guys I have got a good chemistry
  with," Faldo said ahead of the PGA Championship at
  Wentworth. "One (Olazabal) mainly plays in America and
  one (McGinley) mainly plays in Europe. "I felt that was very
  important, for me to have good eyes and ears close to the
  ground, good feedback, and also for the players to know
  lines of communication were now open.
</prop>
</item>
<item id="news02">
...
...
```



**DATA FEED**



**APPLICATION**



**CHANNEL**

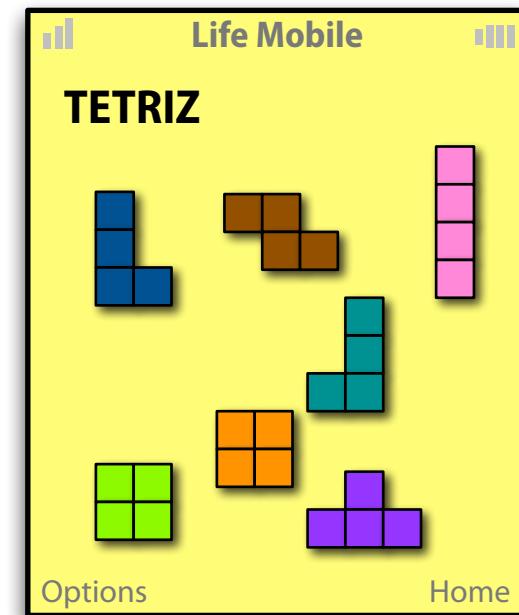
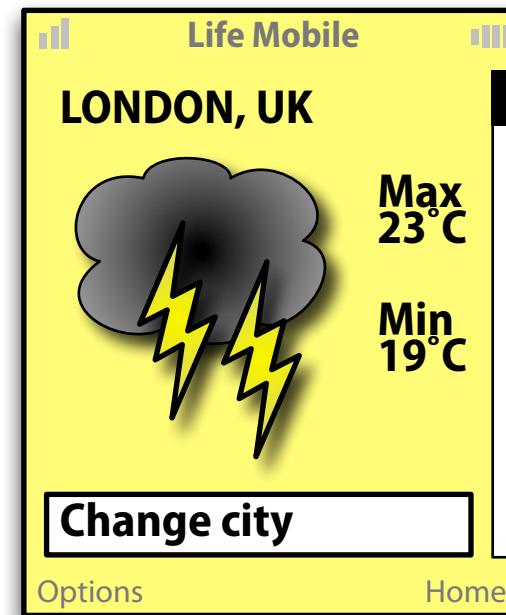
#### Notes:

Flash Cast v1.2 restricts channels to a fixed common size. This size is determined by the operator/service provider with 100K being the reference size.  
Thus, for a reference size channel, the size of the data feed plus the application must not exceed 100K.

# More on channels

- **Channel taxonomy**
- **Application/feed relationship**
- **Simple rules for good mobile content**

### 3 basic types of channel



#### 1. Editorial

Editorial channels typically comprise large volumes of text with some pictures. Normally use a headline selection style of interaction.

Editorial channels should wherever possible use a templated approach where the same application can be used as the basis for multiple channels.

#### 2. Non-Editorial

Non-editorial channels typically have a smaller feed with a graphically richer application.

May require a preference setting UI to allow users to customise the feed according to their own requirements (eg. zodiac sign, current location, choice of stock tickers, etc).

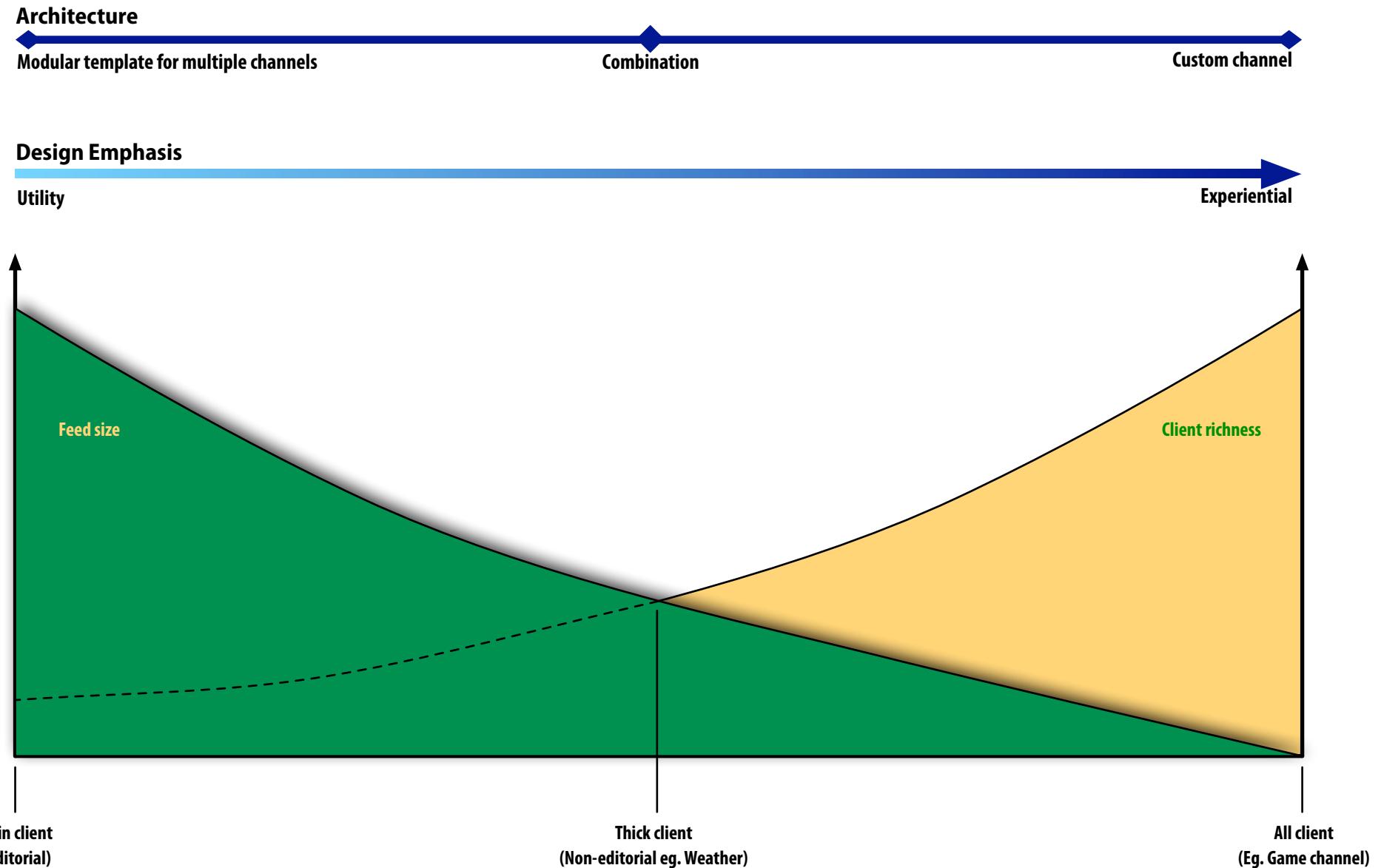
Non-editorial channels will typically be custom designs with the emphasis on the experience.

#### 3. Gaming and others

Gaming channels do not require a feed although a feed could be incorporated for original gameplay.

Game channels can make use of preferences for persisting high scores and other user data.

Channel posture varies according to feed size





[Read more...](#)



## 1. Bite-size

Small screen-sizes, variable lighting conditions, and a high likelihood of distractions mean that mobile information should be presented in small, easily consumable, portions.

Editorial articles and other channel content should be limited to easily consumed bite-size chunks. However content should not be artificially constrained to comply with this guideline. It is equally important that content is self-contained (see 2) as easily consumed.

## 2. Self-contained

A channel should provide a complete user experience in itself. Information presented to the user should, as far as possible, be independent from other information sources.

External linking should be used with restraint, and not relied upon to complete article or channel items. "Read more..." links leading to WAP/web pages, should not be a common feature of Flash Cast channel construction.

## 3. Not time critical

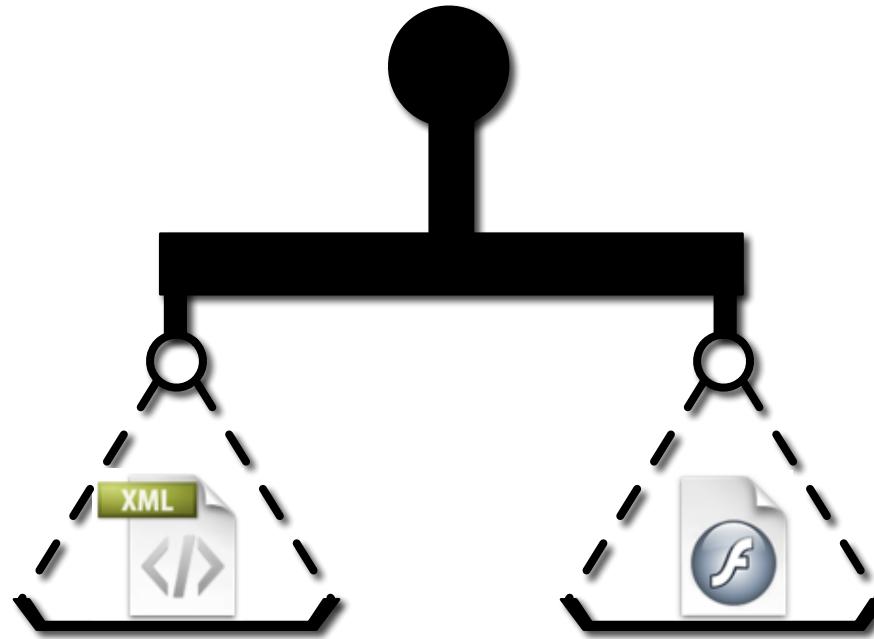
The nature of Flash Cast channels is such that they are generally updated at set regular intervals. Furthermore, mobile content consumption and network coverage can often be sporadic, content must be organised to take these factors into account.

Whenever time-critical information is used in a channel, the user interface should provide a clearly visible time stamp for that information.

# About feeds

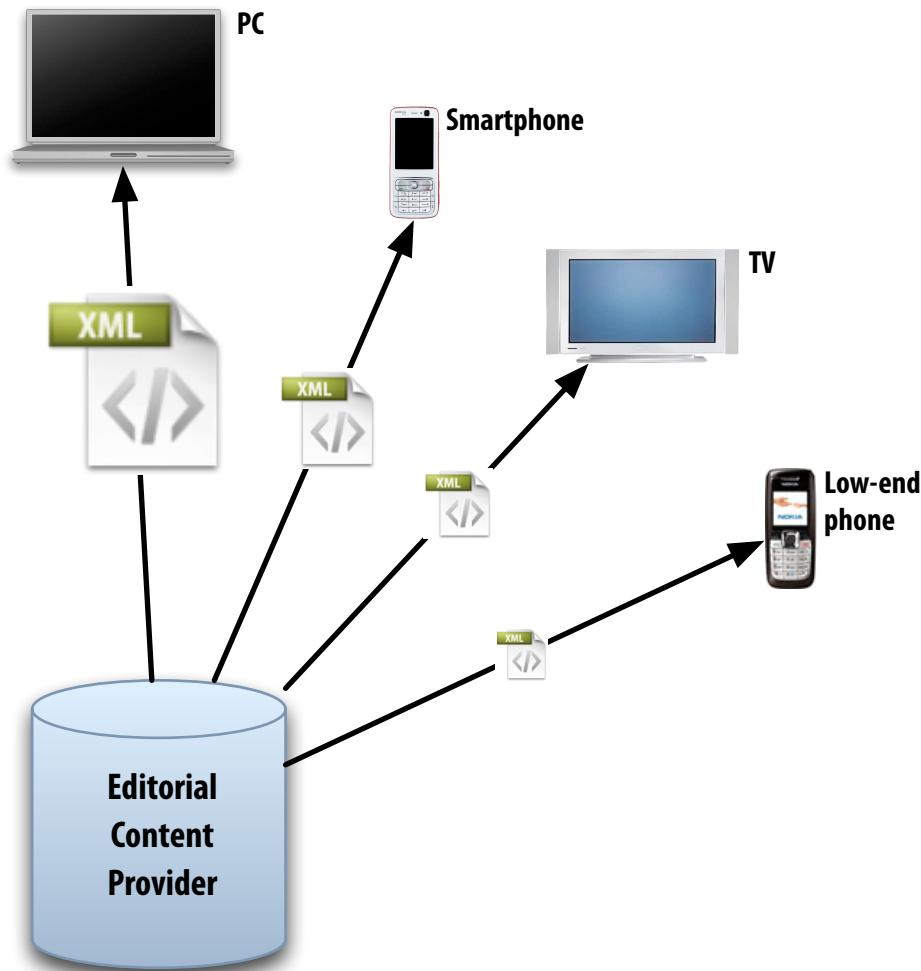
- **Importance of feed quality**
- **Editorial vs Non-editorial feed differences**
- **Multi-platform authoring**
- **Feed considerations**
- **Update frequency vs. Lifespan**
- **More feed considerations**

Feed quality is equally important as application quality



**Flash Cast cannot make bad content look good!**

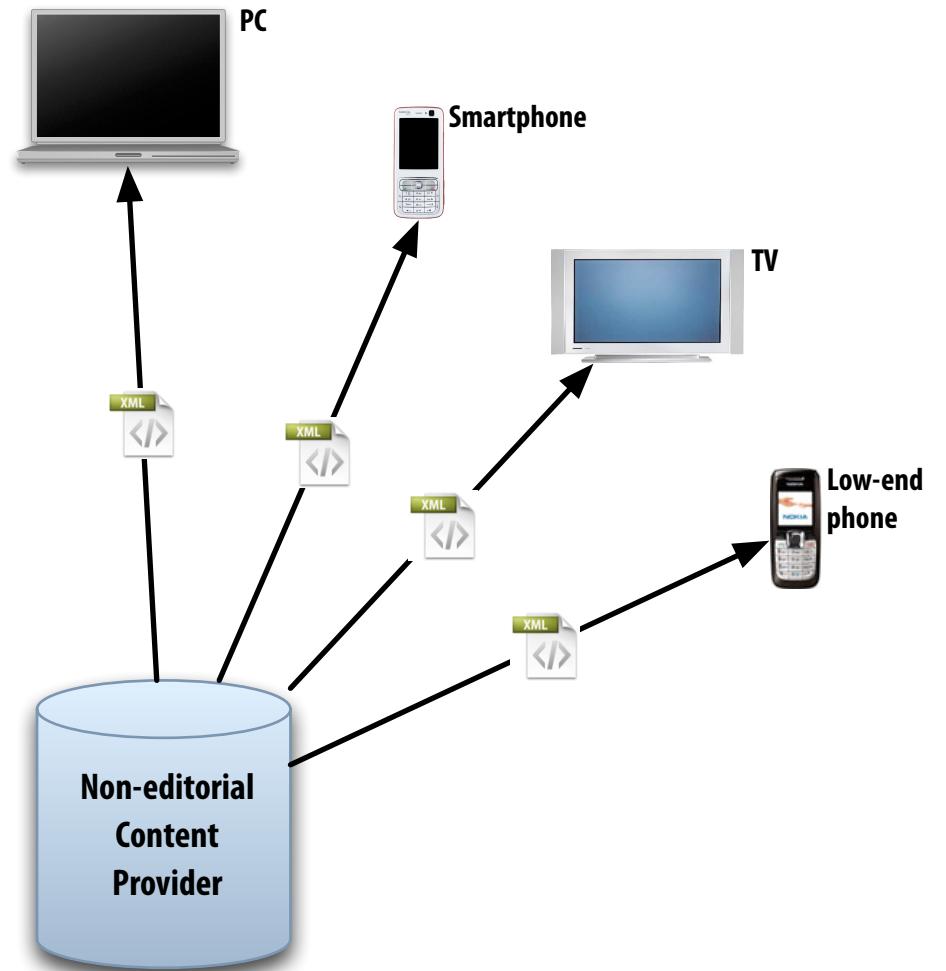
## Editorial content must scale to suit target device



### Editorial content

Editorial content needs to be scalable so that is appropriate to the capabilities of the receiving platform.

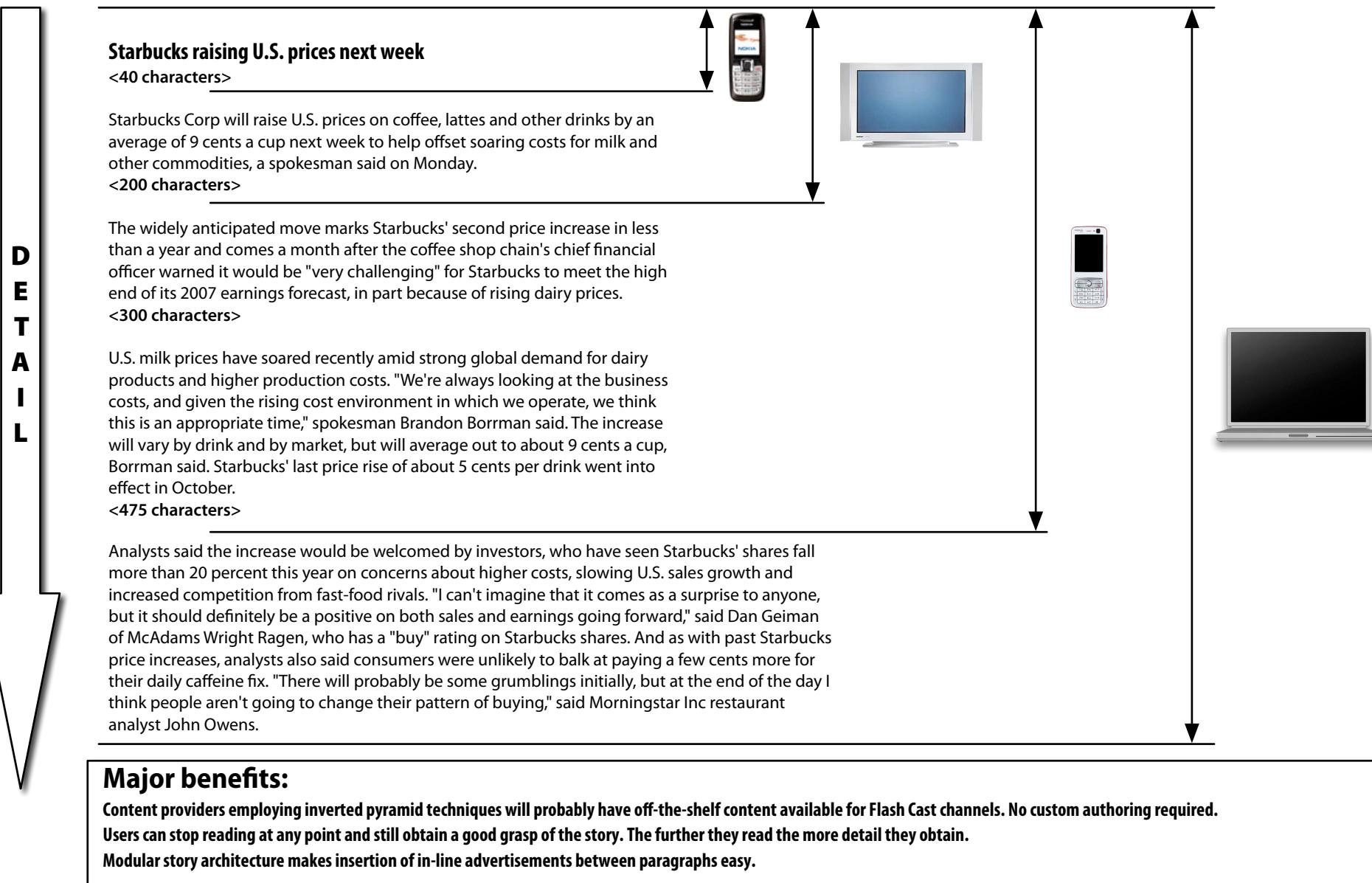
This requires the content provider to be pursuing a multi-platform strategy and using 'inverted pyramid' journalistic techniques.



### Non-editorial content

Eg. Weather forecasts, stock prices, etc  
Should remain constant and largely independent of the capabilities of the receiving platform.

## Inverted pyramid journalistic techniques - typical news story





## 1. Images

To avoid making channels too text heavy ideally feeds should contain images. Content providers should be able to provide these.

Content providers may not be able to supply images at a specified size. Some resizing may have to be performed by the Flash Cast server. There may be Copyright restrictions that apply here.

Images may be embedded in the feed as binary information or referenced via URL. Embedding images in the feed is superior as it eliminates the need for the image to be fetched for inclusion.



## 2. Feed format

Ideally feeds should be provided in a standards based format as proprietary formats may cause problems or make integration more difficult.

Flash Cast can work with any text-based feed but it is easiest to work with an XML based feed.

Make sure that the feed supplier has experience of supplying finished feeds to end users and not just raw feeds to content redistributors for further editing.

Some content providers may have restrictions on editing (eg. truncation) of the content they supply.

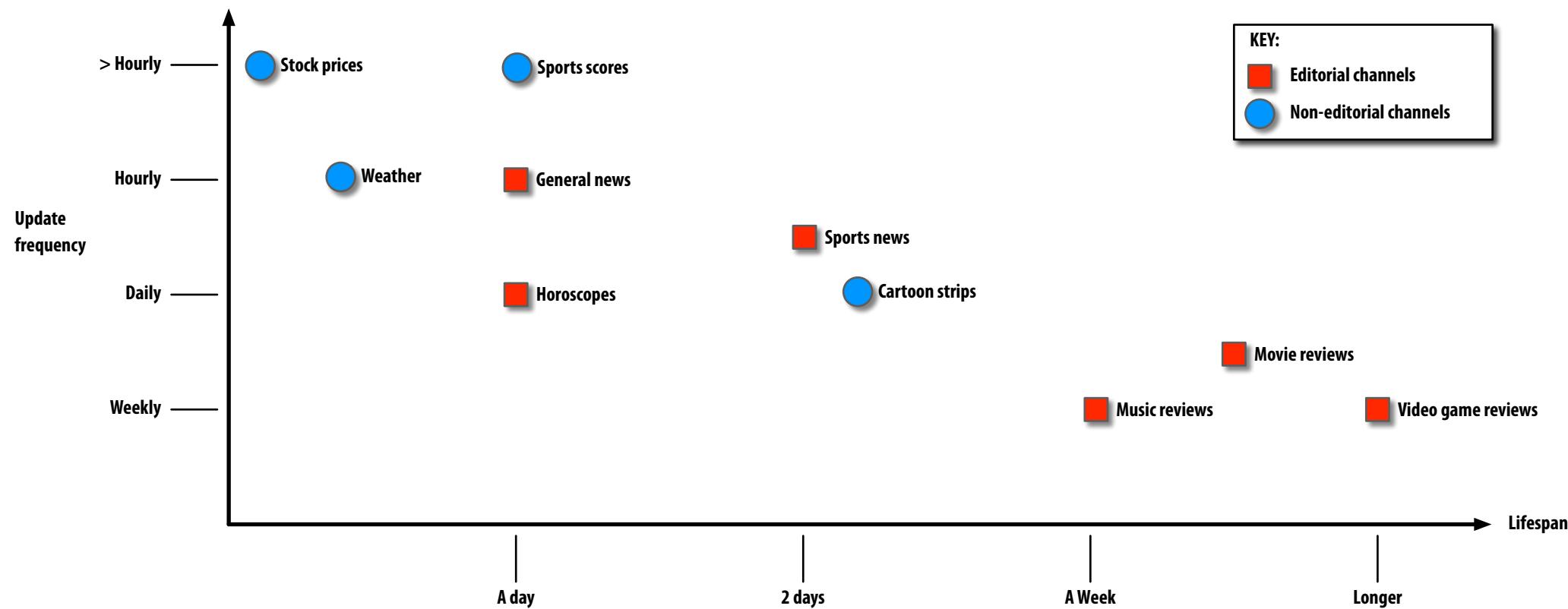


## 3. Update frequency and availability

If content isn't updated at an appropriate frequency then this can result in a poor or broken user experience.

See "Update frequency vs Lifespan" slide for details.

Feeds have widely varying update frequencies and lifespans

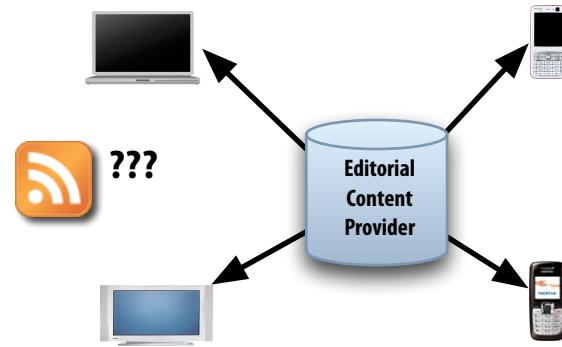


### Implications:

Content may need to be cached or otherwise persisted to ensure continued availability when update frequency is low.

Merging feeds from different sources (eg. Sports news + Sports scores) may be problematical owing to differing update frequencies

Channels that are only updated on a weekly basis may appear lifeless and can fall into disuse. Better to trickle updates out rather than use a single weekly update.



### 1. Content providers

RSS feeds can be repurposed for use within Flash Cast channels. However it is unlikely that a raw RSS feed will be ideally suited to Flash Cast distribution.

The level of detail contained in the feed may not be sufficient and the assumption that a browser is available for providing access to the complete article is wrong for Flash Cast. The user should not have to invoke the browser to fully consume an item.

Some content providers may not be prepared to create a special feed specifically for a Flash Cast channel. Look for partners that are already creating content for multiple platforms as this will increase the likelihood that they will have an existing content feed that is right for Flash Cast provisioning.



### 2. Application/Feed compatibility

The channel application and channel data feed are two parts of a whole. Thus it is important that the XML schema for a given feed is designed in consultation with the application developer (and vice versa) to ensure an optimal experience.

Feed content must be structured in such a way so as to satisfy the requirements of the application modules. For example if the application design includes an accordion module the corresponding feed must include accordion headers as well as specific accordian content to be specified in the feed.

Modular, templated approaches help to minimise the variation between individual channel feeds.



### 3. Involve users

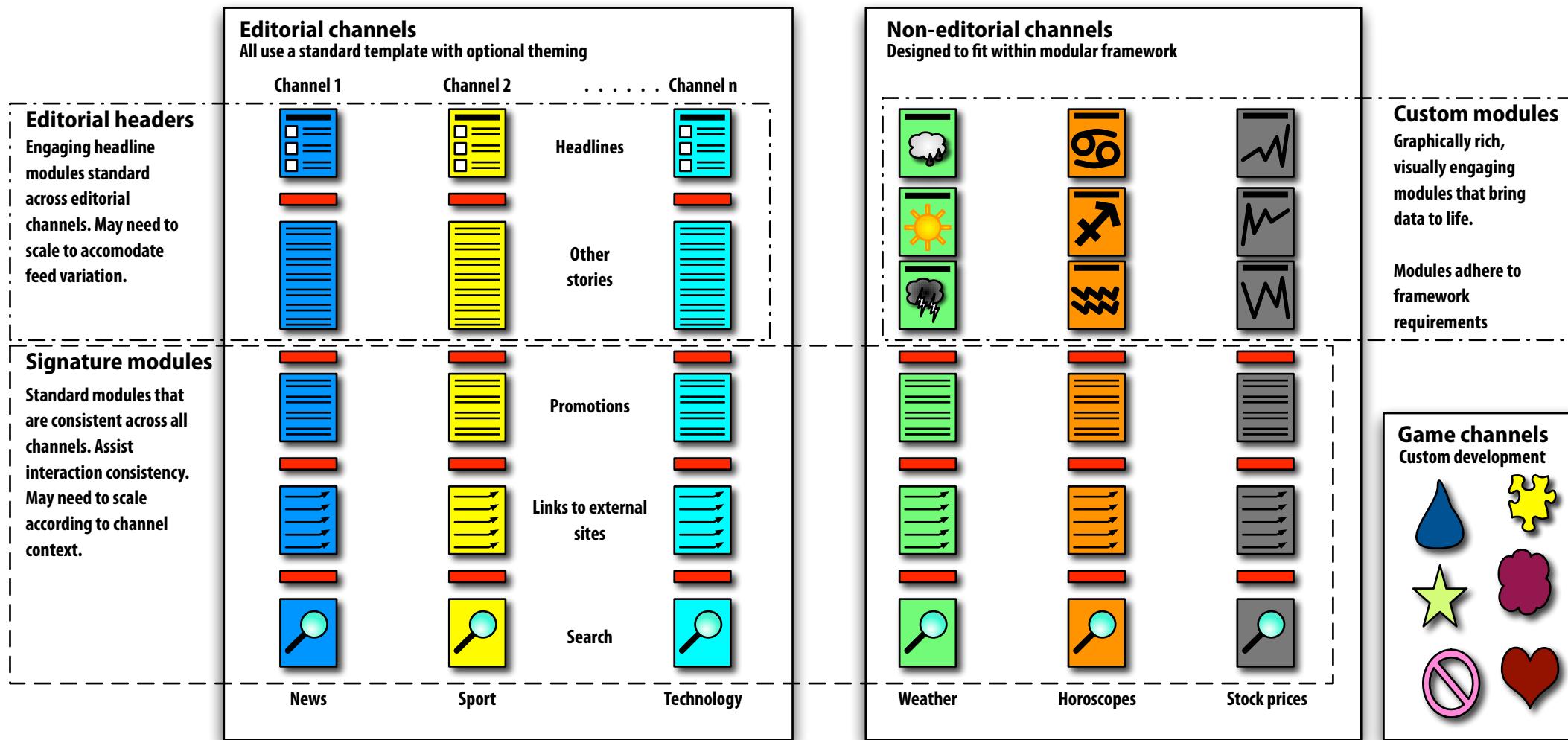
Involve users in the design and development process. Find out what their needs are and how these can be met with a channel offering.

Find out what are the users needs for mobile data and modify feed structure accordingly.

# Architecture and design principles

- **Modular architecture**
- **How much development?**
- **Homogenised schema 1**
- **Homogenised schema 2**
- **Reuse of modules**
- **Modular approach explained 1**
- **Modular approach explained 2**
- **Benefits of a modular architecture 1**
- **Benefits of a modular architecture 2**
- **Benefits of a modular architecture 3**

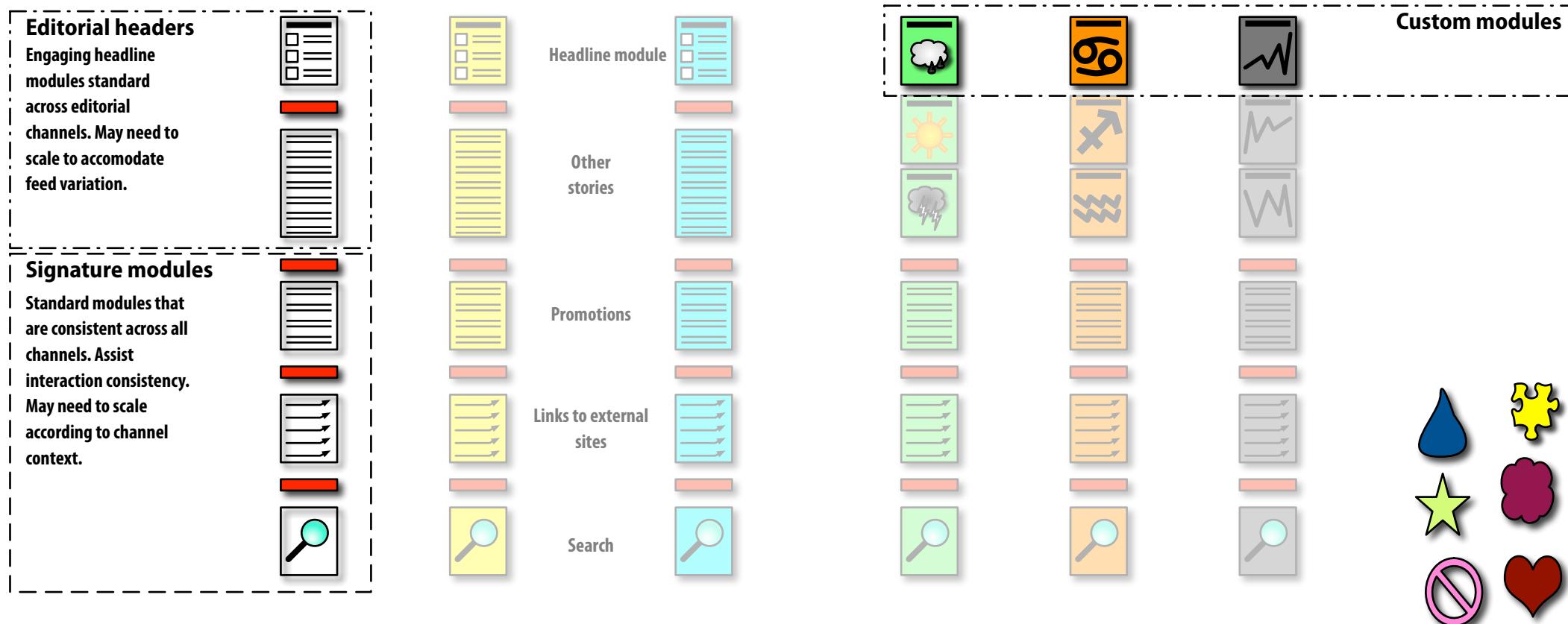
A modular architecture provides numerous design, development, testing and user benefits



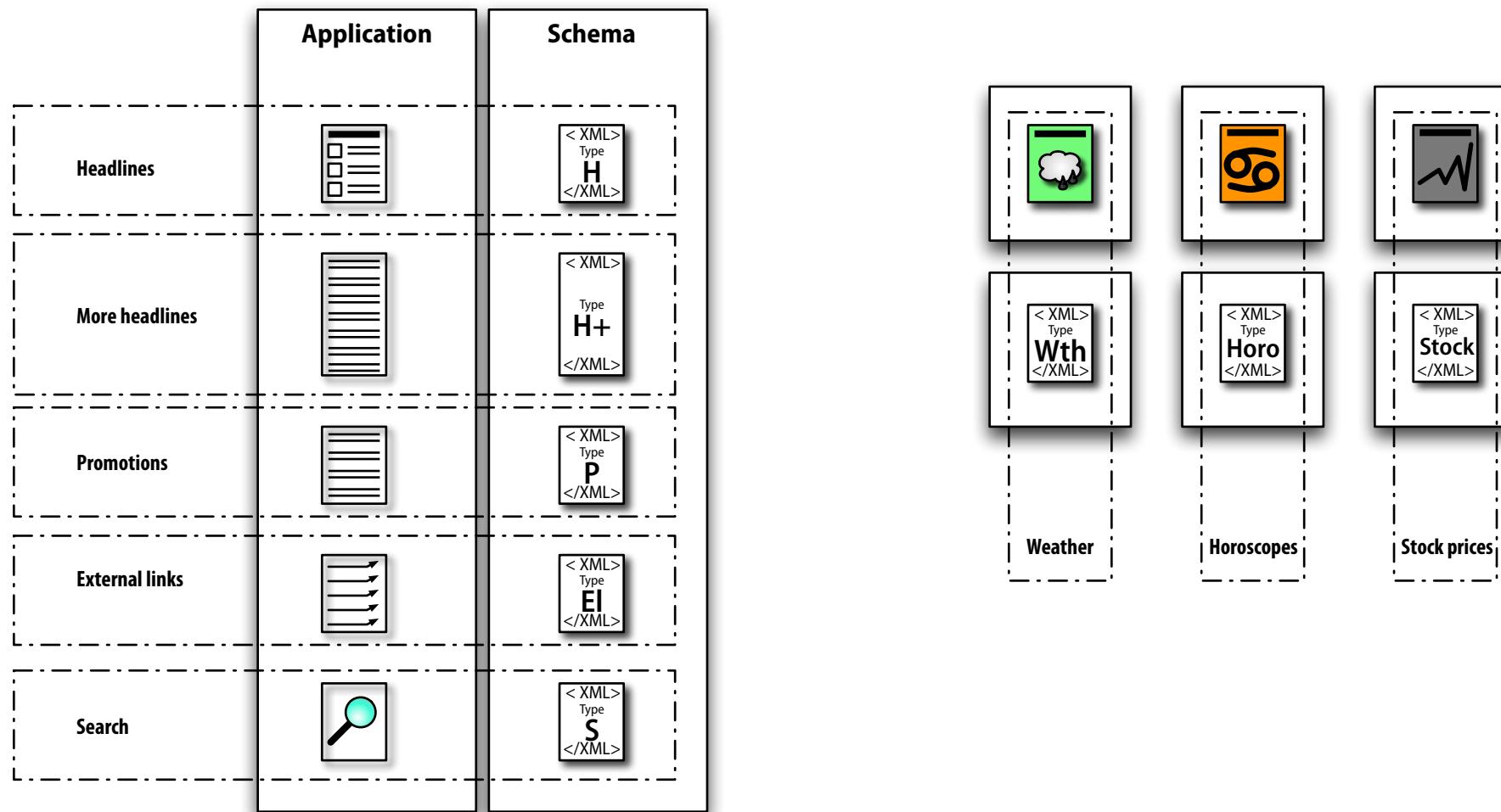
= Inline advertisement.

See <http://www.mmaglobal.com/mobileadvertising.pdf> for advertising guidelines.

Modular architecture means that surprisingly little development is required.



A modular schema that suits all of the data driven channels in a channel package



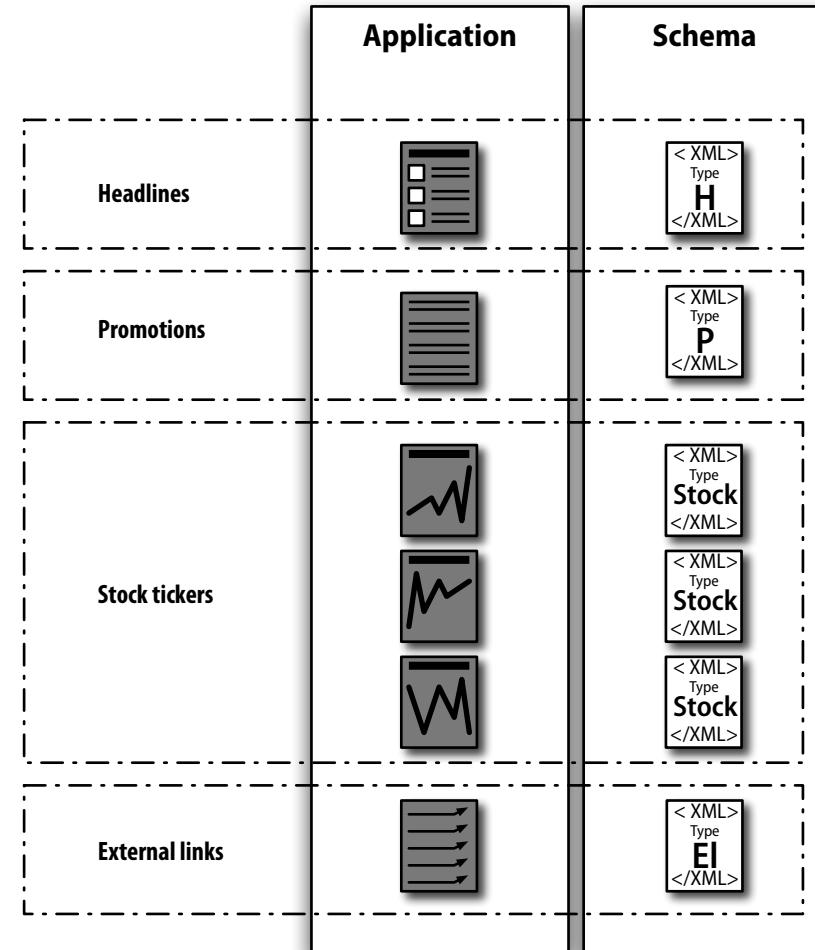
## Notes:

An application (comprising a number of application modules) plus a schema (comprising a number of schema modules) is a template.

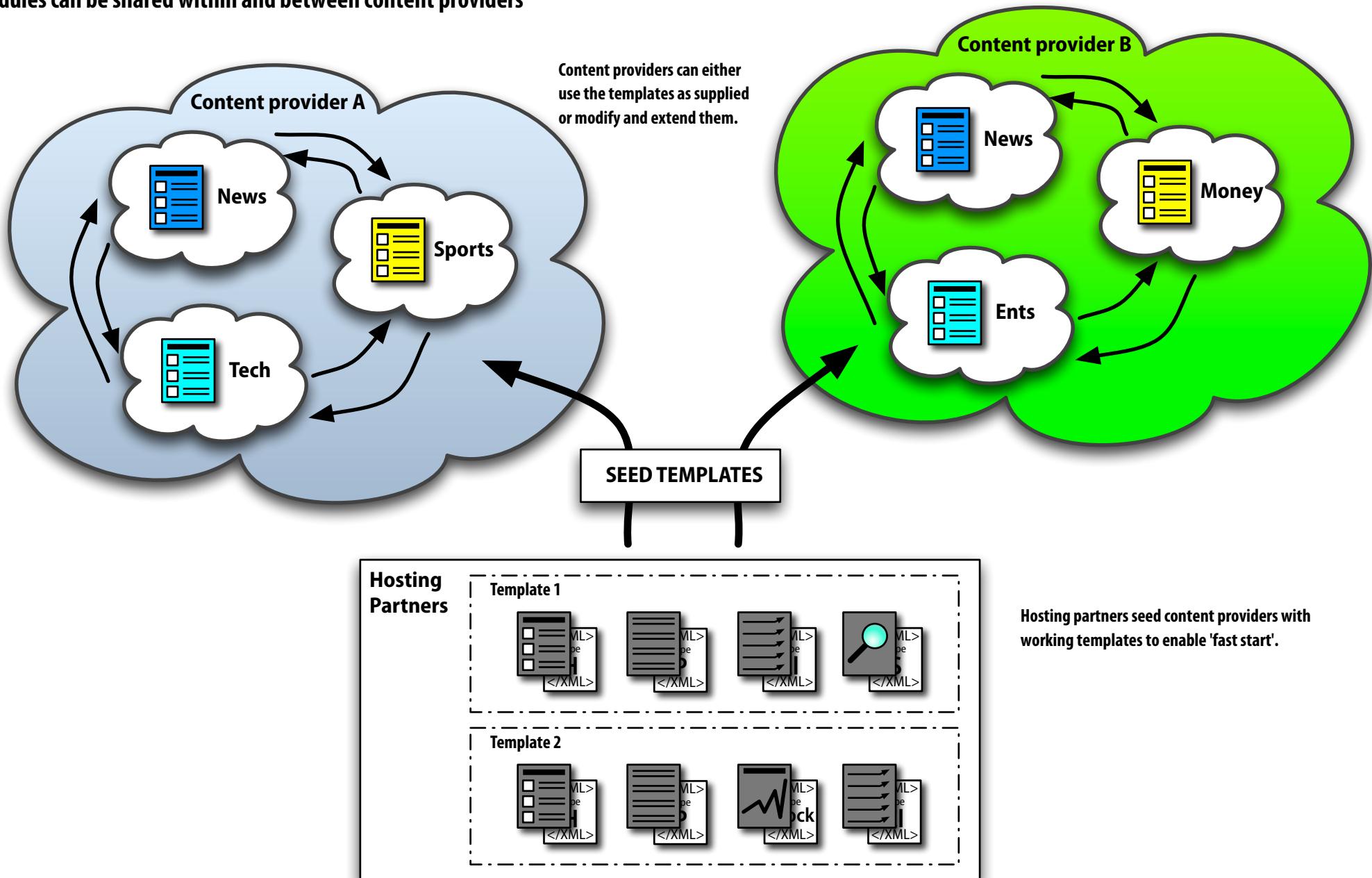
The modular approach decouples the application from the schema. This allows individual modules in either the application or the schema to be modified without affecting the entire channel.

The modular arrangement makes it easy to construct custom channels

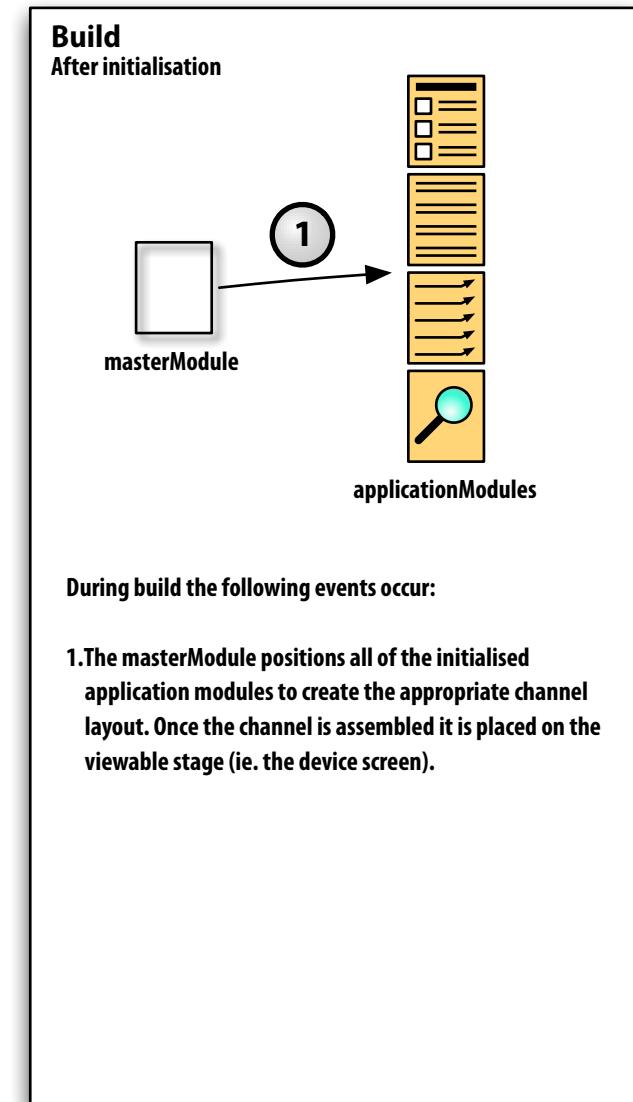
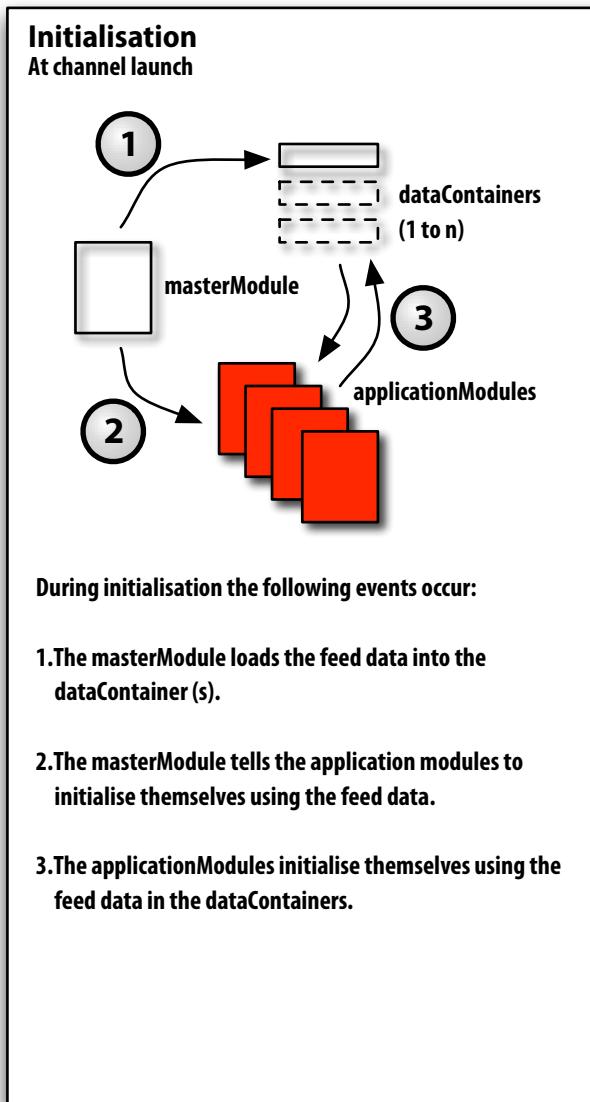
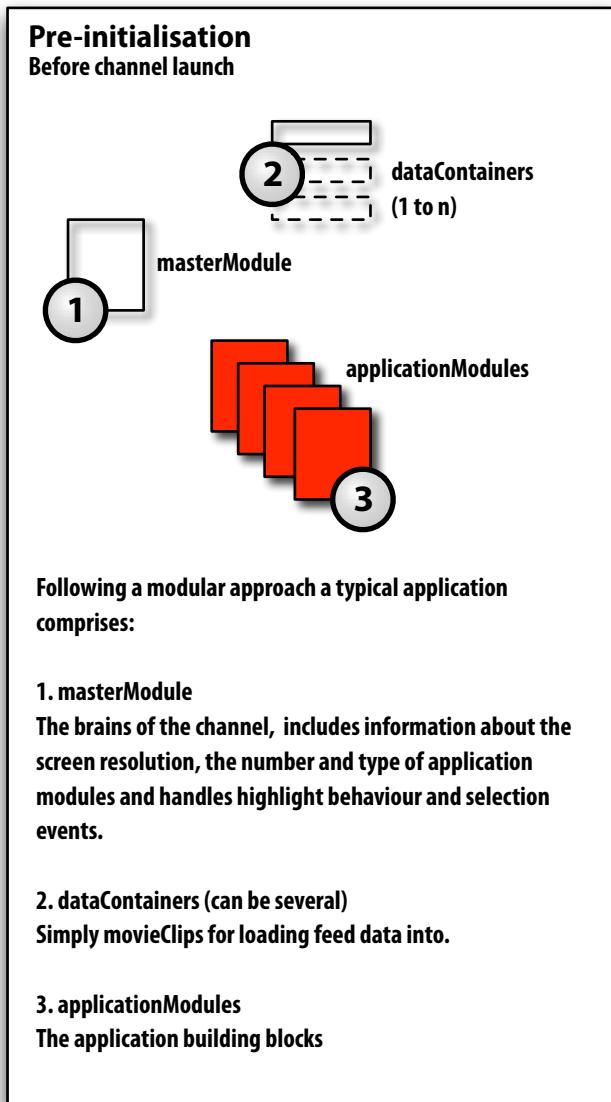
New channel design  
(Hybrid editorial/non-editorial)

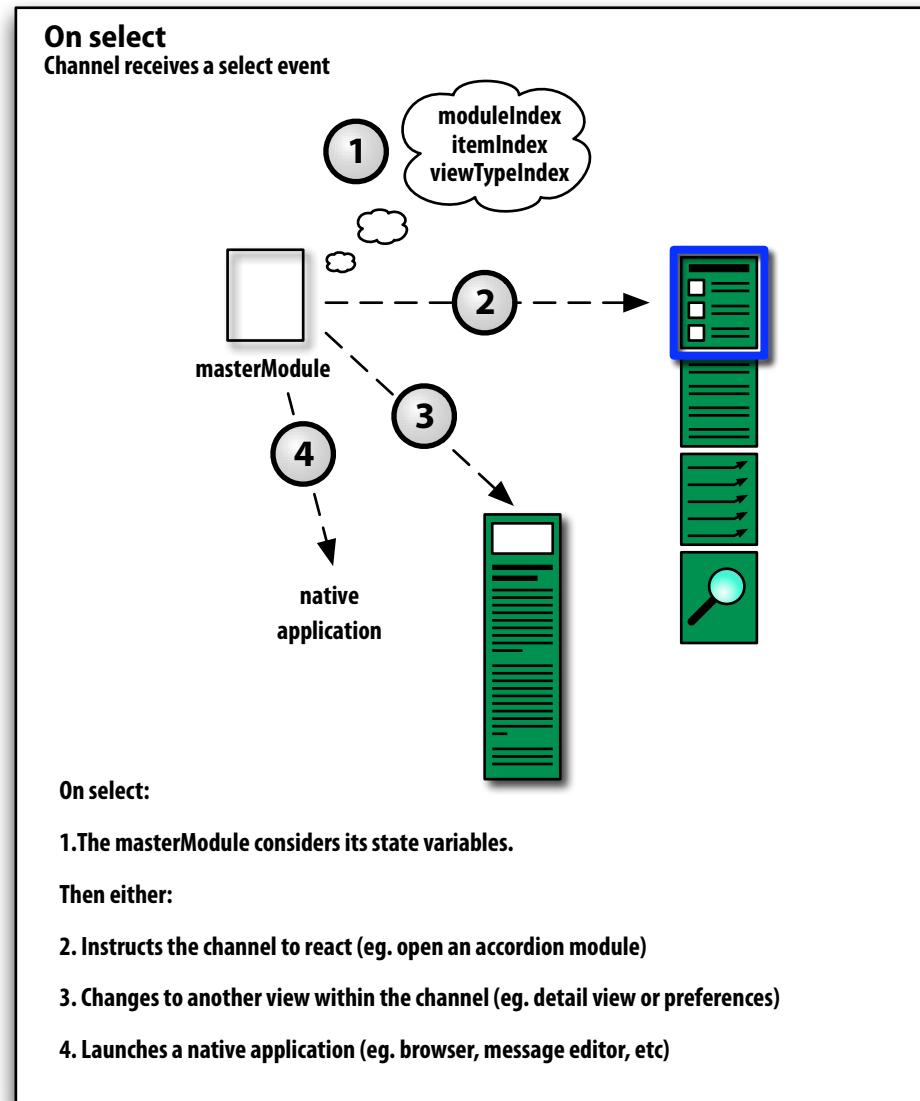
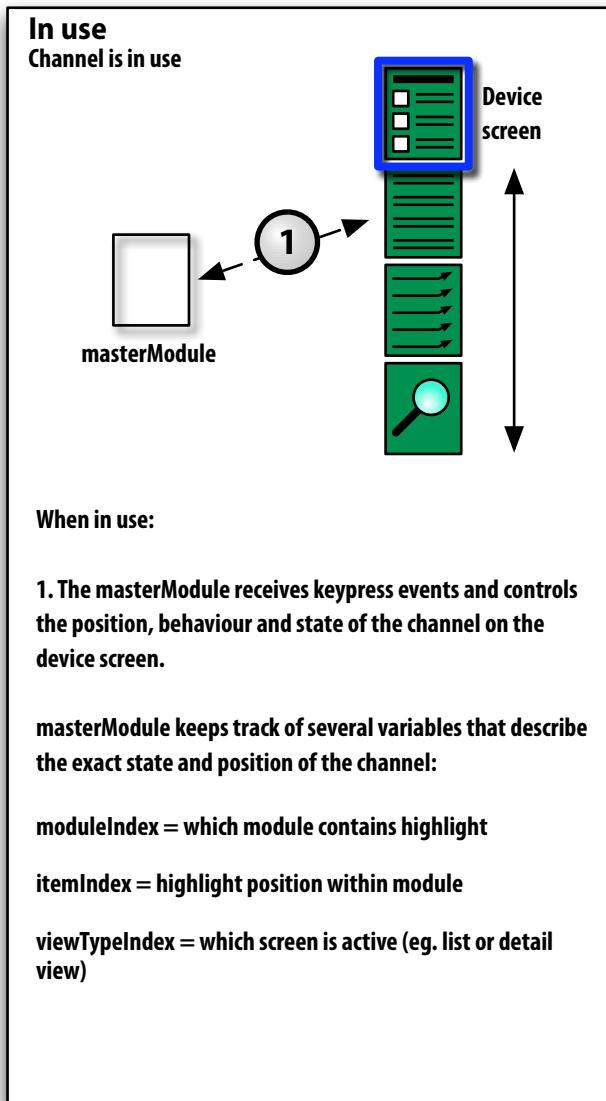


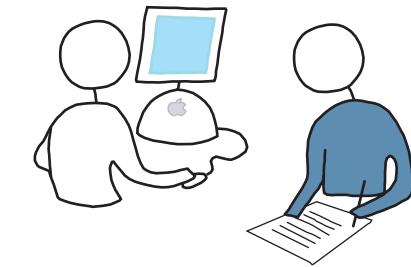
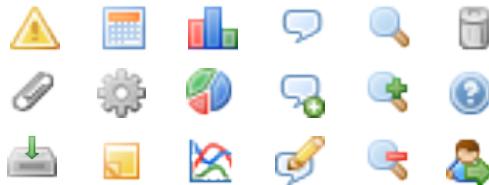
## Modules can be shared within and between content providers



## Schematic introduction to modular channel operation







## 1. Design benefits

A modular approach reduces the design load through standardisation. Design effort can be concentrated on creating robust reusable designs rather than firefighting multiple design issues with multiple channels.

## 2. Development benefits

Adobe can provide a number of standard modules to assist in rapid development of initial channel offering.

Custom modules that fit into the modular framework can be easily developed.

Changes to individual modules can be made without the entire channel being affected. Modules can be reordered without changing the schema and a minimum of changes to the application.

## 3. Testing benefits

Standard modules will have already been subjected to rigorous release testing.

Custom modules can be easily tested in isolation prior to use in channel construction and deployment.



### 4. User benefits

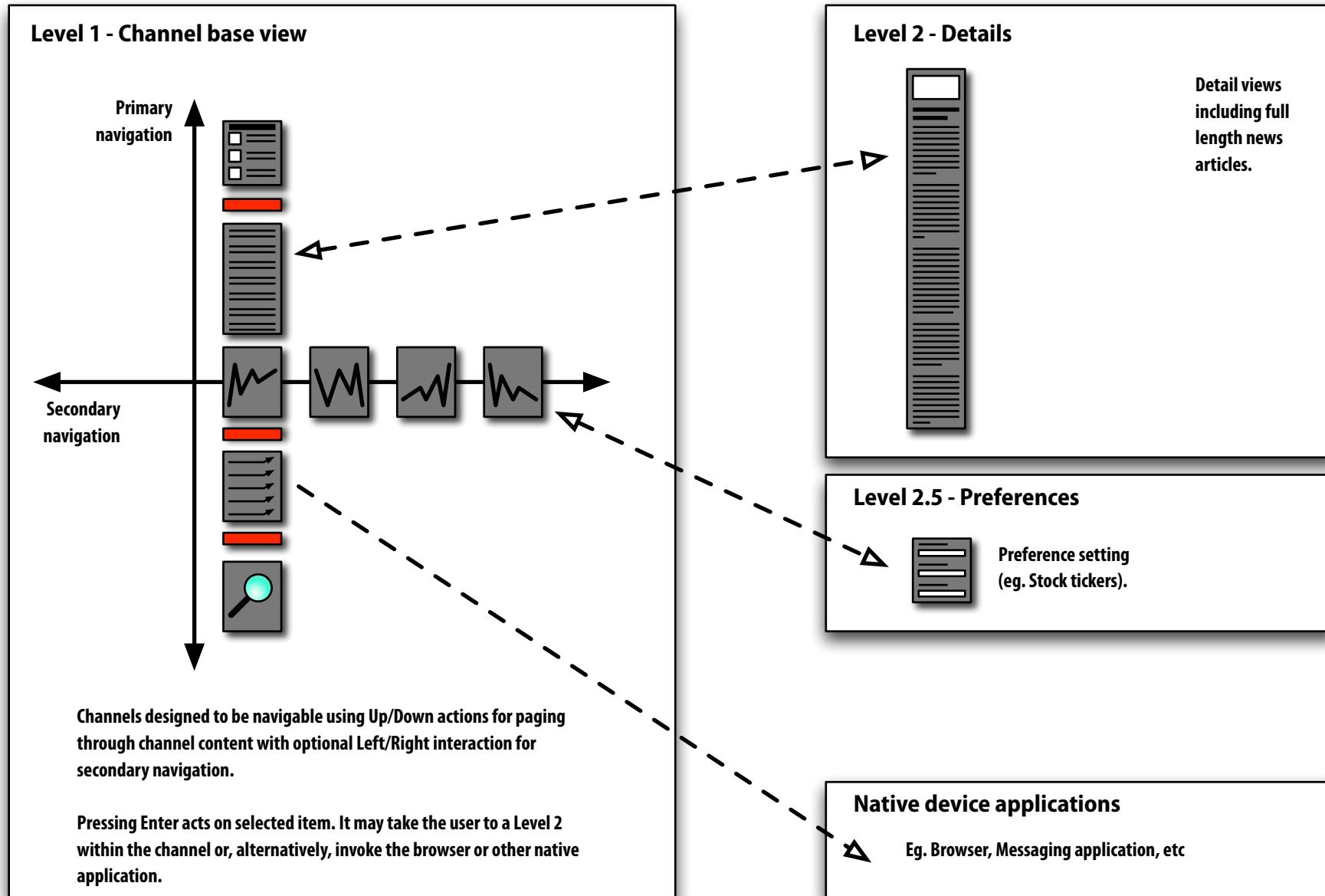
User is not required to learn a unique interaction pattern for every channel. A consistent navigation model allows the transfer of learned interaction behaviour between channels.  
(See following slide).

### 5. Client benefits

The combination of design, development and testing benefits contribute to an efficient and streamlined project delivery.

User benefits contribute to increased overall customer satisfaction and increased service uptake.

### Consistent navigational model - Assists usability, user only needs to understand a single interaction model



# Design patterns

- **Content paging**
- **Headline selection**
- **Accordion component**
- **Shelf and carousel**

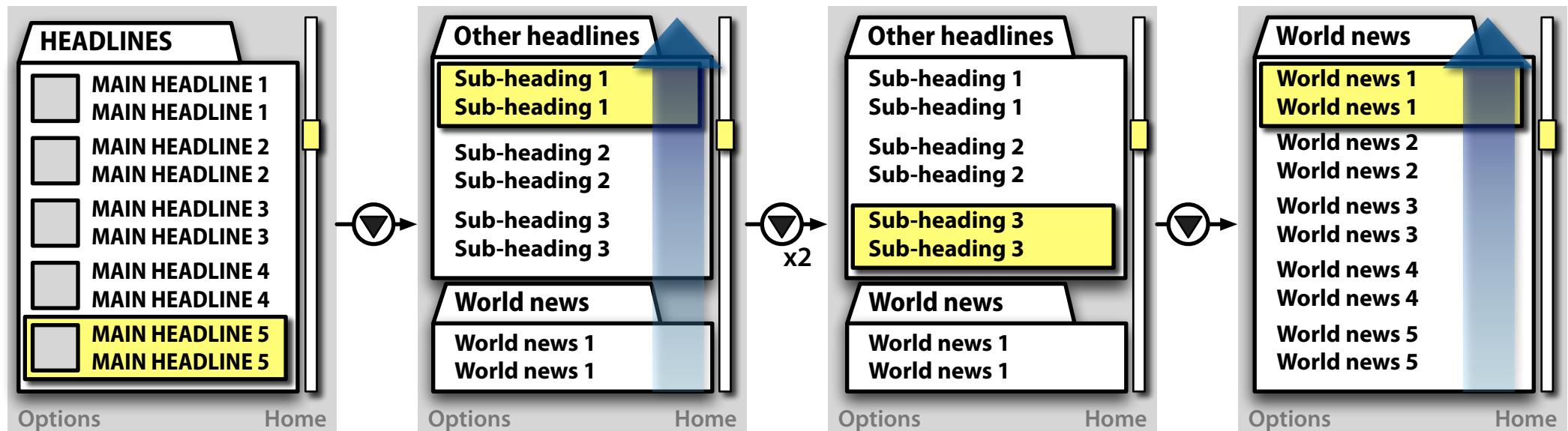
Each channel comprises a series of 'pages' arranged in a column.

Moving the highlight off the bottom of one page moves the first item of the next page to the top of the screen.

Each page has a clean appearance with a distinct heading to effectively show where it begins. No messy half lines visible

This paging interaction makes for a straightforward up/down interaction model that can be applied consistently across channels and the user doesn't have to learn a new interaction style for each channel.

Each page can have unique responses to other key events. For example, left/right key events could be used for driving a tabbed style interaction within one page and a carousel style interaction in another. Similarly the enter key can be used for expanding content in a page containing an accordian or for switching to the detail view in a page using headline selection.



## First page

Shows primary content in an appropriate presentation format.

Scrolling off the bottom of the page animates the second page of content to the top of the screen.

## Second page

Highlight moves to the first focussable element in the new page of content.

Note that page headers, such as the World news header, provide cues for upcoming content.

## Second page cont.

User scrolls page content as normal.

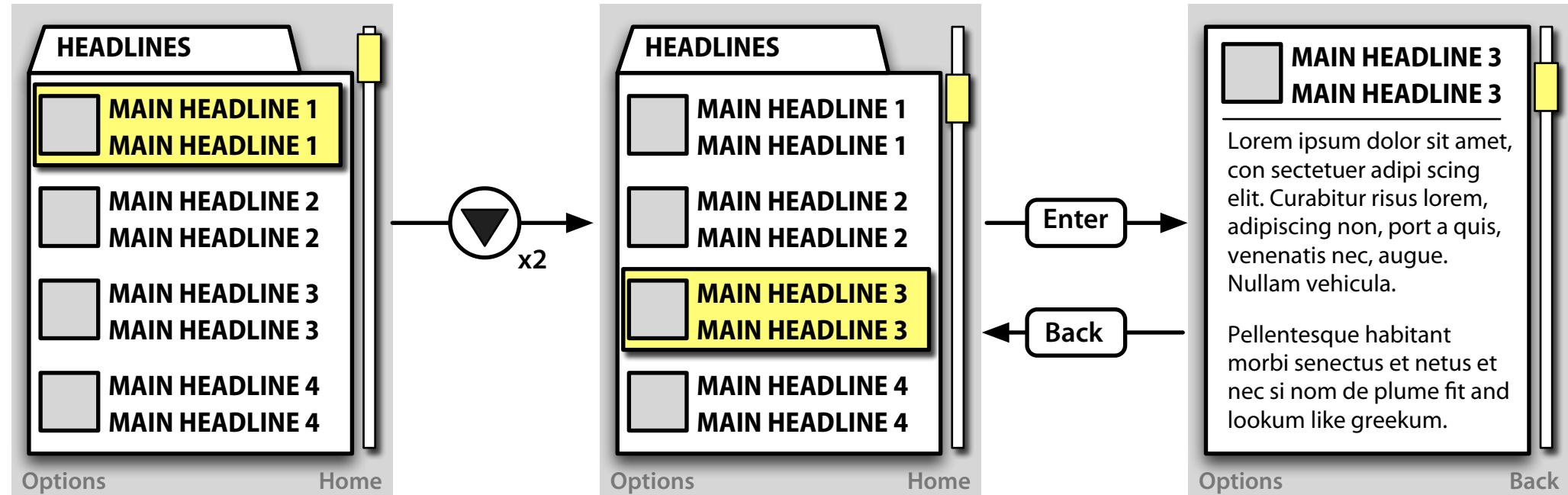
Scrolling off the bottom of the page highlights the first focussable item in the new page of content and simultaneously smoothly animates it to the top of the screen.

## Third page

Highlight at top of new page.

Paging behaviour ensures that there is normally a heading at the top of the screen.

Enables familiar list/detail style interaction



Scrolling highlights a block of text, pressing Enter presents the entire article

Pressing 'Back' softkey returns user to the list of headlines.

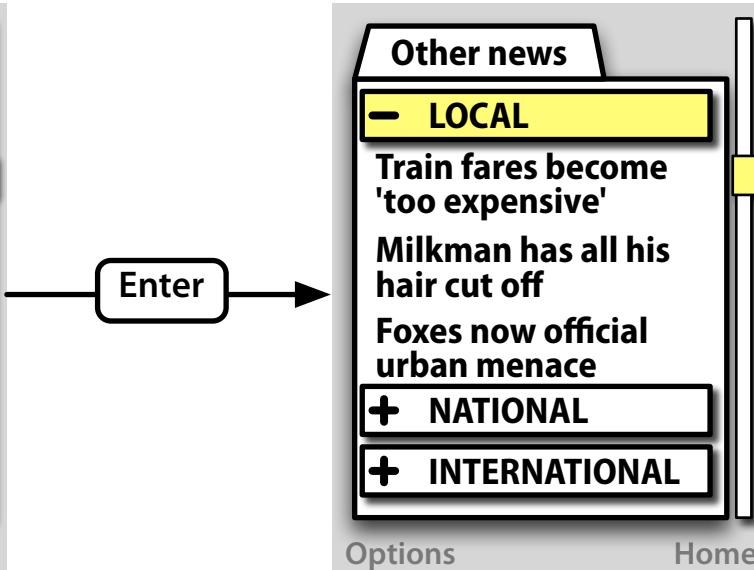
## Accordion component

v1.5 updated on Fri Apr 04 2008

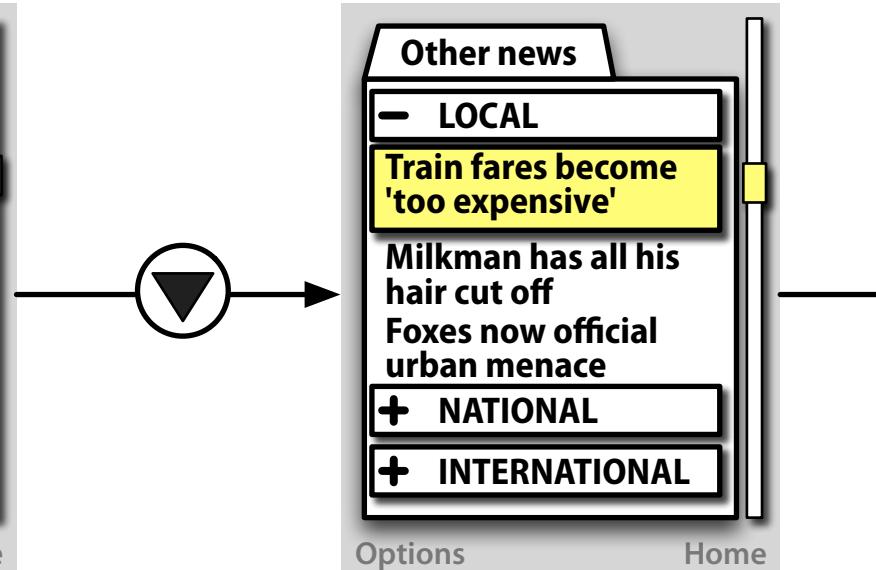
Useful component for condensing large numbers of grouped headlines into a small space.



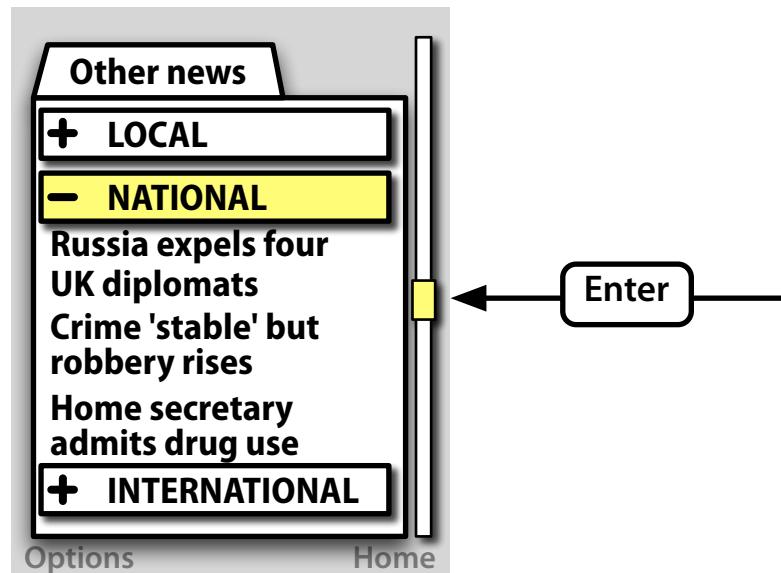
Compact accordion component takes up minimal screen estate.



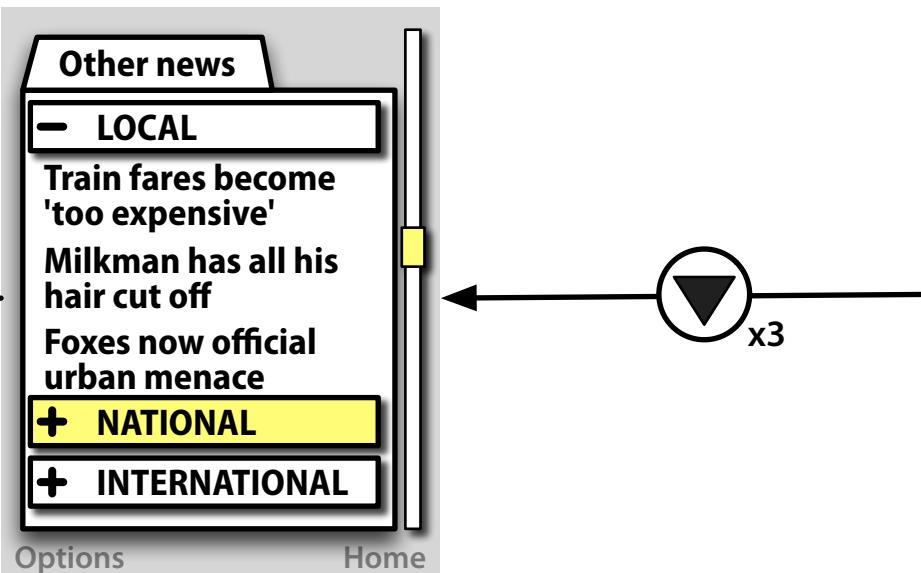
Pressing enter expands the highlighted accordion item revealing content



Content can be easily navigated.



Opening a second accordion item automatically collapses the first.



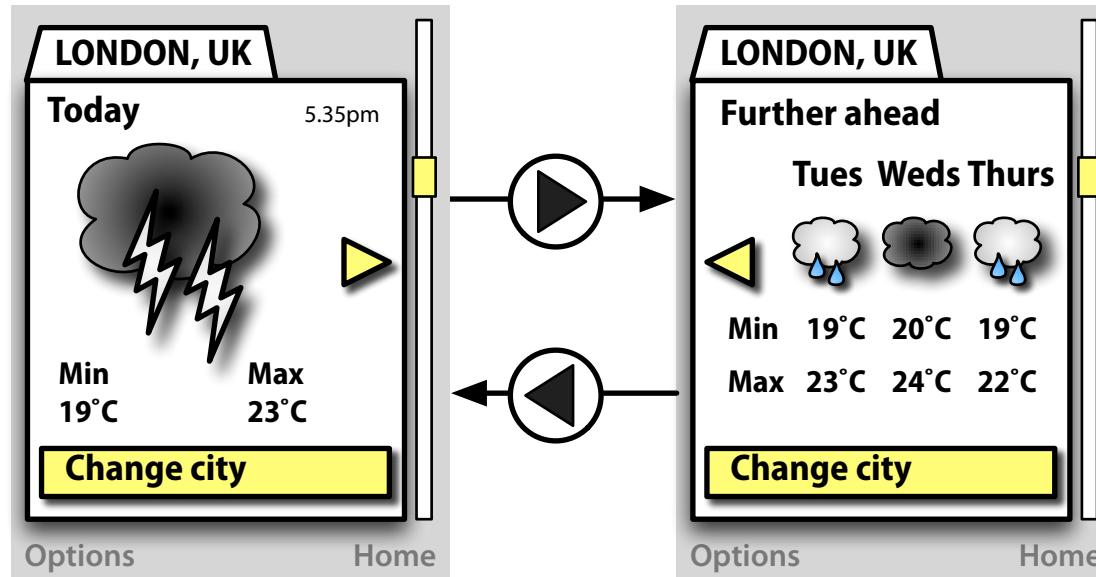
Flash Cast - A guide for content developers

## Shelf and carousel components

v1.5 updated on Fri Apr 04 2008

### Shelf

Useful component for providing additional information with minimal navigational disruption.



Shelf uses secondary navigation to provide more information without disrupting user navigation in the primary direction.

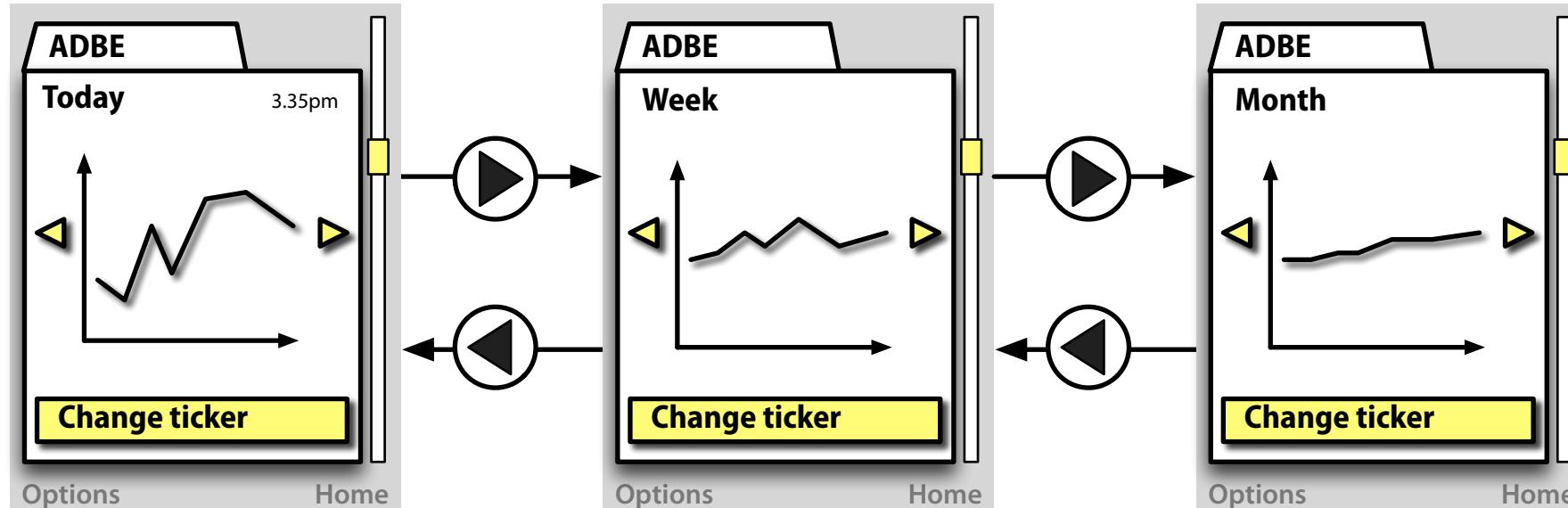
The user can continue to page content up and down independently of the state of the shelf.

Carousel also uses secondary navigation to provide alternative views without disrupting user navigation in the primary direction.

The user can continue to page content up and down independently of the state of the shelf.

### Carousel

Alternative to Shelf when more detail is required.



# On fonts

- **Available fonts**
- **Font usage tips**

### The Flash Cast client supports a mix of pixel/raster and vector/non-raster fonts

#### Pixel/raster fonts

standard 07\_55

The quick brown fox jumps  
over the lazy dog

**standard 07\_65**

**The quick brown fox jumps  
over the lazy dog**

standard 09\_56

The quick brown fox jumps  
over the lazy dog.

For pixel/raster fonts to look crisp they need to be used at their native point size (8pt) and to be aligned exactly on a whole pixel. Misalignment or using them at a non-native size will result in a blurred appearance.

Because of their small size these pixel fonts may not be suitable for body text on hi-resolution devices (QVGA+) which normally have a very small dot pitch.

#### Vector/non-raster fonts

Myriad Web Pro

The quick brown fox jumps  
over the lazy dog

Myriad Web Pro

The quick brown fox jumps  
over the lazy dog

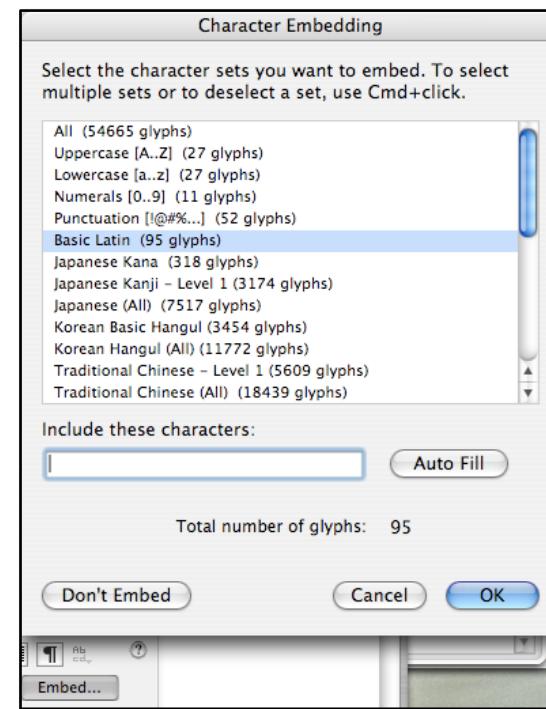
Myriad Web Pro

The quick brown fox jumps  
over the lazy dog

The Myriad Web Pro font is best used at sizes over 12pt.

Again for best results vector fonts should also be aligned on a whole pixel. Misalignment will result in a distorted appearance.

```
4
5 fscommand2("SetQuality", "high");
6
```



## 1. Playback quality must be set to high

Font appearance can appear degraded if the playback quality is not set to high.

To set the quality to high just include the following actionscript on the first frame of the movie:

```
fscommand2("setQuality", "high");
```

## 2. Embedding glyphs

Individual glyphs can be embedded in applications using the Embed feature in the Flash IDE.

This technique should be used extremely sparingly as it adds considerably to application sizes.

Numerals are a possibility as they typically only require an additional 10 glyphs to be embedded.

# Designing for different orientations

- **Target resolutions**
- **Viewing non-native resolution content on device**
- **Available screen estate**

QVGA is the minimum target resolution. Devices may feature screens in different orientations.



Landscape - Nokia E61



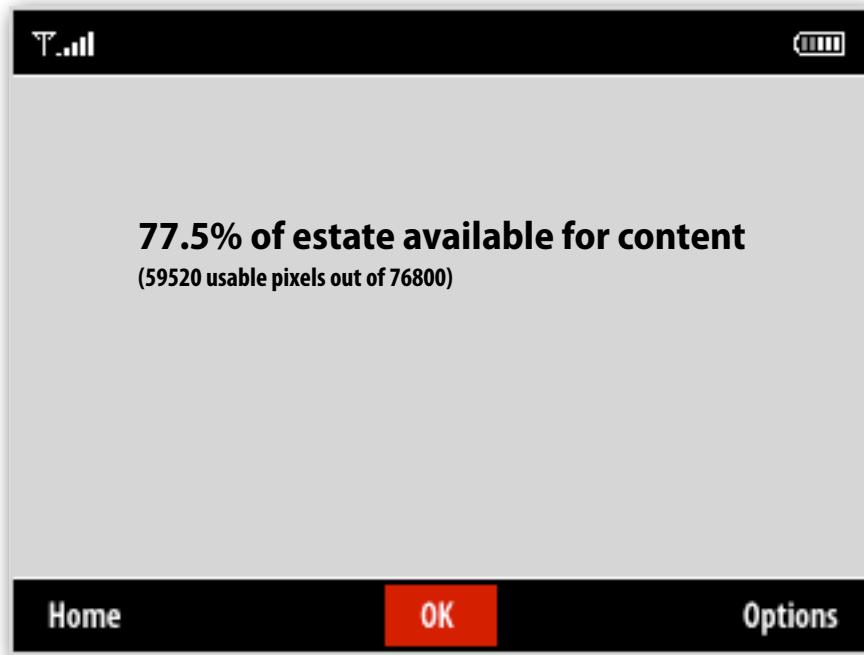
Portrait - Nokia N95

Content automatically scales proportionally to suit resolution of viewing device without distortion.

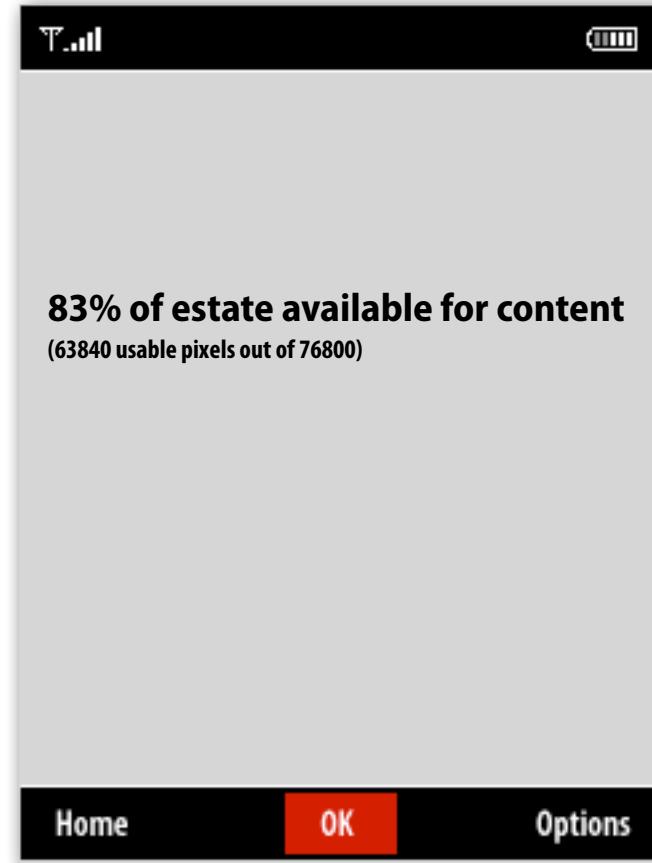


Portrait - Nokia N95

More screen estate is consumed by chrome on landscape devices



QVGA - landscape

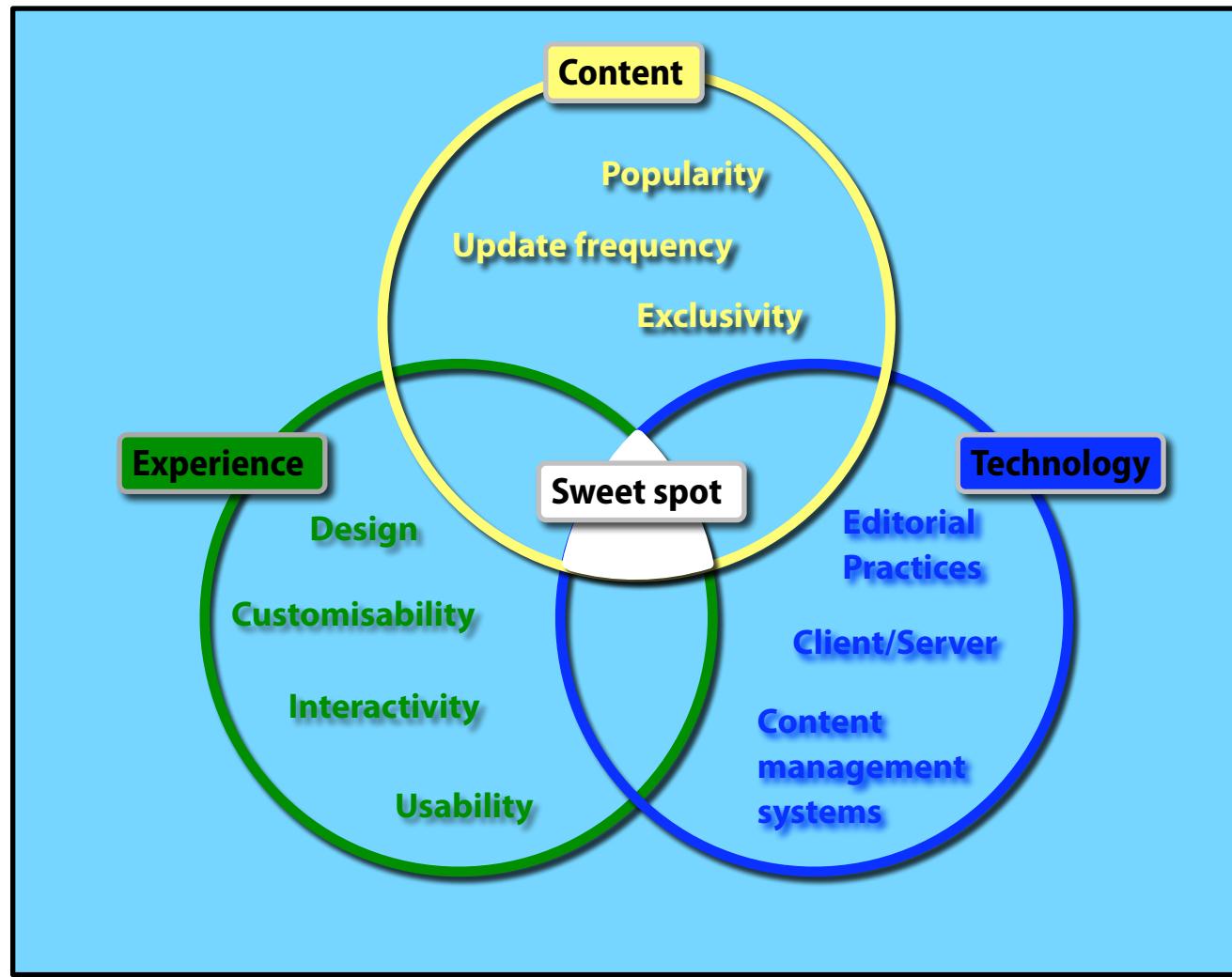


QVGA - portrait

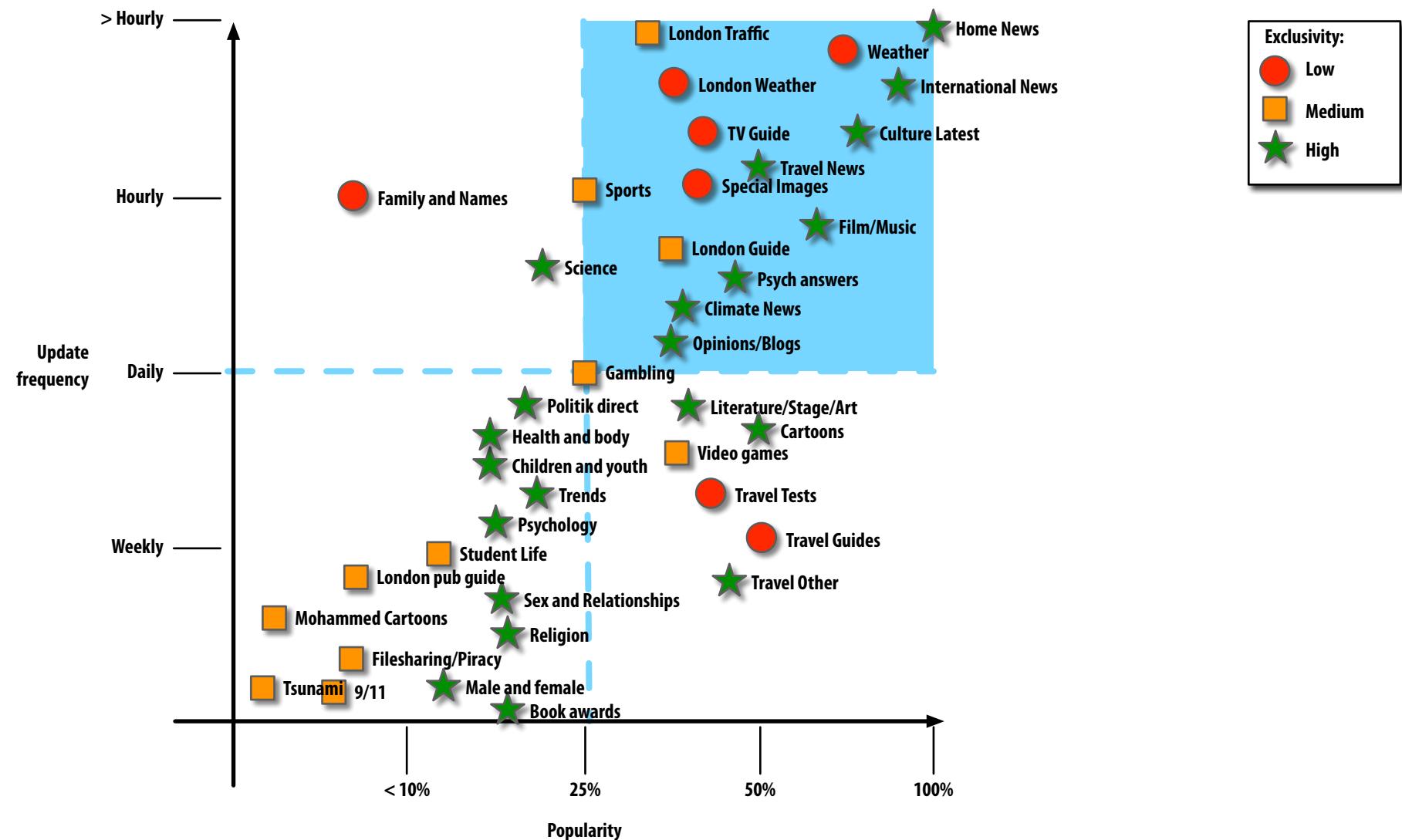
# Choosing content for Flash Cast

- **Creating a great experience**
- **Identifying content candidates**
- **Channel offering lifecycle**
- **Why Flash Cast?**
- **Importance of Partnership**

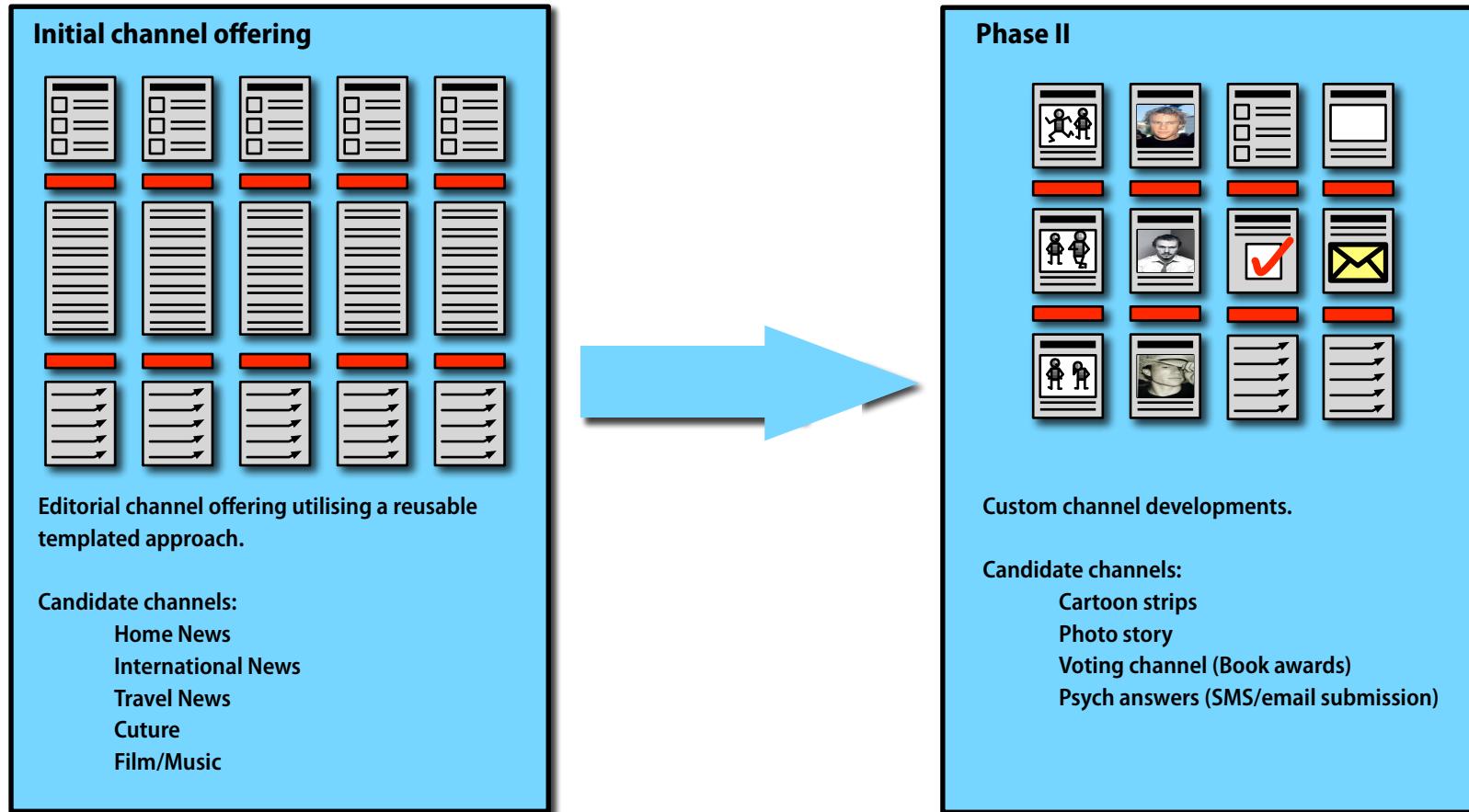
Identifying and delivering a good Flash Cast channel offering entails the union of three separate components



Candidate content needs to be popular, updated frequently, and preferably, exclusive



Initially launch editorial channels followed by custom channel developments





## 1. Fresh and modern

Flash Cast can compliment a content providers web presence by bringing a classy new look and feel to it's mobile offering.

**Look, because Flash Cast enables the use of rich graphical presentation of content.**

**Feel, because Flash Cast allows the use of animations, visual transitions and build effects that create an engaging and rewarding content experience.**

## 2. Fit for purpose

Flash Cast has been designed for mobile deployment. It provides a unique mobile-centric experience that is immediately familiar to mobile phone users.

**It utilises the capabilities of the device to provide a unique highlight-based navigation experience which is far superior to existing mobile web experiences.**

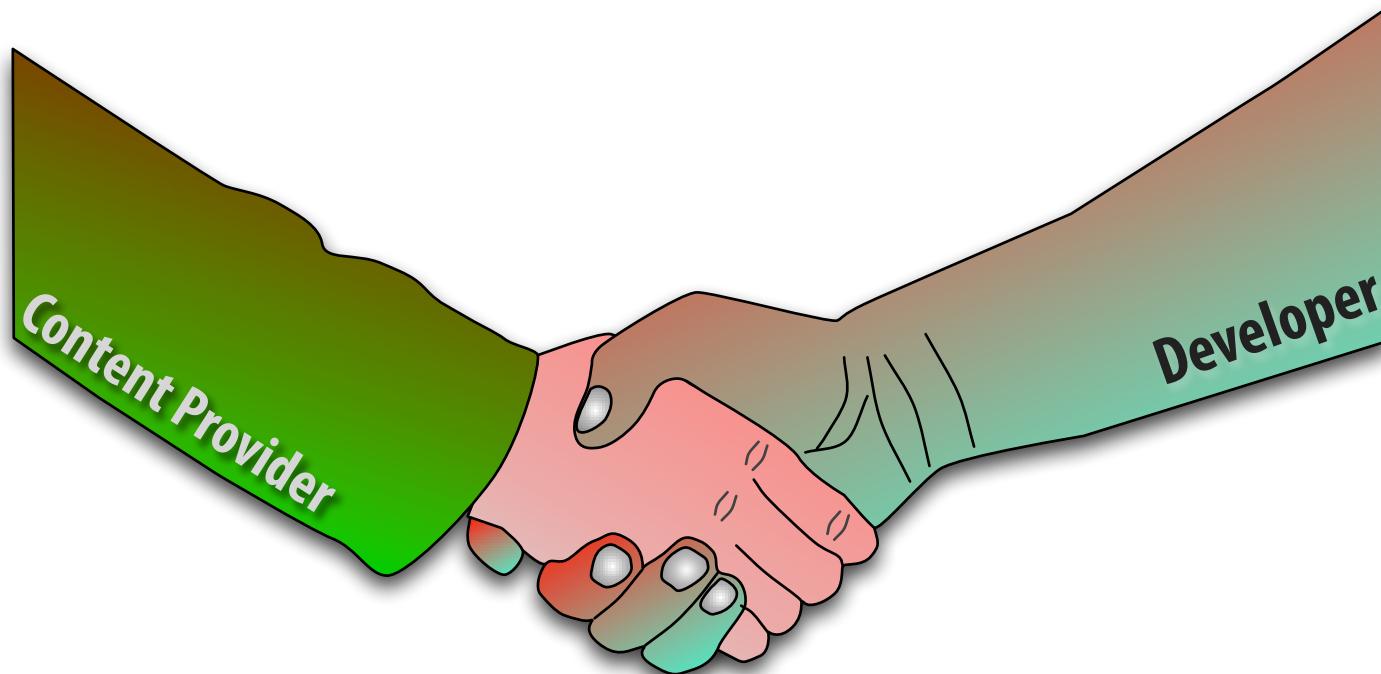
**Flash Cast is infinitely customisable allowing individual experiences to be crafted that presents content in the best possible way.**

## 3. Device integration

Reader engagement is supported with easy and immediate accessibility to device capabilities - SMS, MMS, e-mail and telephone functionality can be easily integrated into the content experience to encourage participation.

Furthermore, the data path back to the content server can be used for gathering user data for use in polls or other interactive experiences.

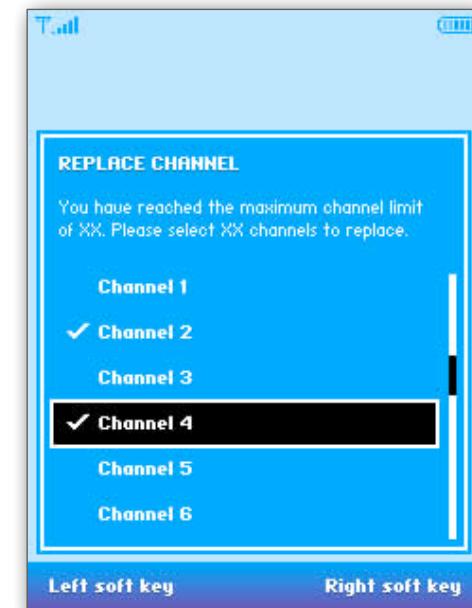
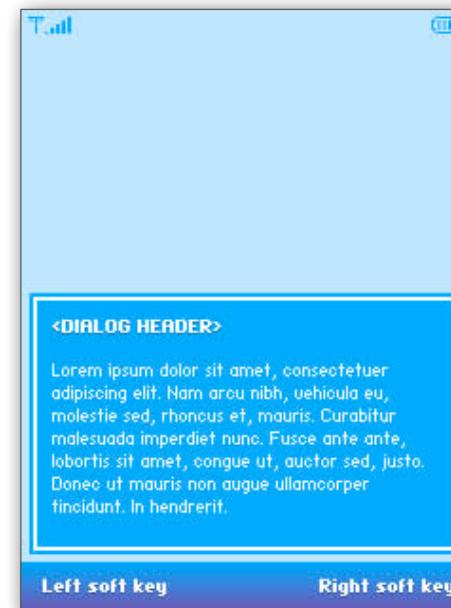
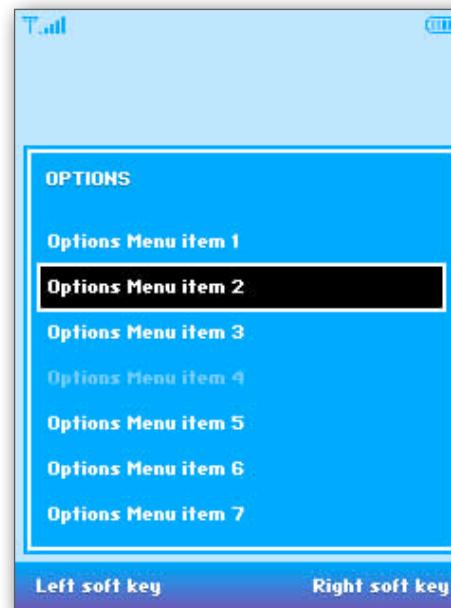
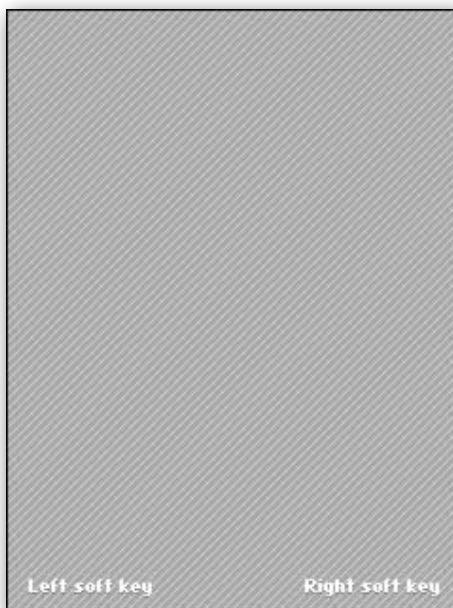
You can't do it on your own



# System UI

- **Common elements 1**
- **Common elements 2**
- **Common elements 3**
- **Customising the system UI**

## Customisable views, dialogs and other elements used across the Flash Cast experience



### 1. Softkey labels (fc01)

Responsible for displaying softkey labels over all other channels. Note background is transparent. Individual channels are normally responsible for providing softkey background with suitable contrast.

Individual channel developers should be informed about size of reserved softkey area and text colour so these can be accommodated in channel designs.

Center softkey may also be optionally be labelled.

### 2. Options menu (fc02)

Responsible for presenting context sensitive menu commands across channels.

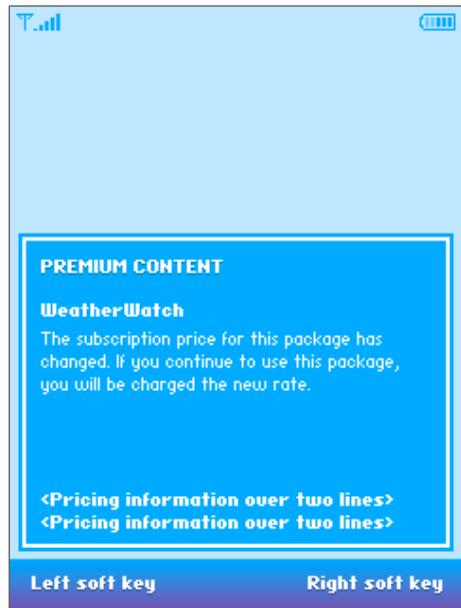
### 3. Basic dialog (fc03)

General purpose dialog.

### 4. Replace dialog (fc04)

Dialog presented when number of subscribed channels exceeds the maximum permissible.

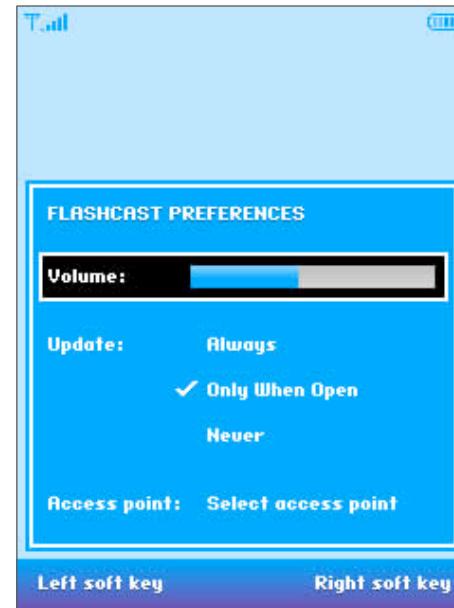
## Customisable views, dialogs and other elements used across the Flash Cast experience

**5. Premium content (fc05)**

Presented when a channel subscriptions is not free.

**6. Channel guide/browser (fc06)**

Used for managing channel subscriptions.  
Normally comprises multiple screens.

**7. General preferences (fc07)**

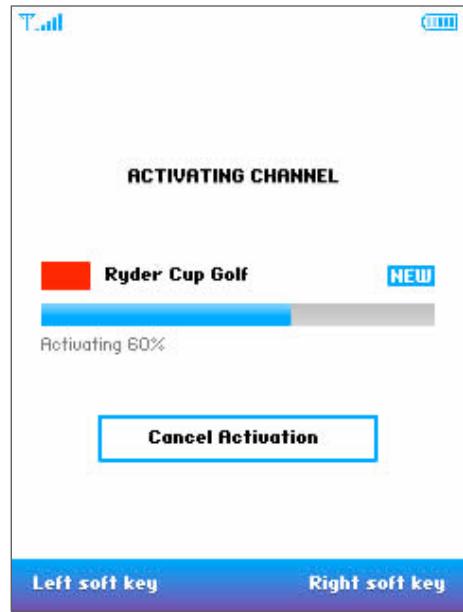
For setting global preferences.

**8. Now playing**

Provides access to currently subscribed channels.  
The equivalent of a home screen for Flash Cast.

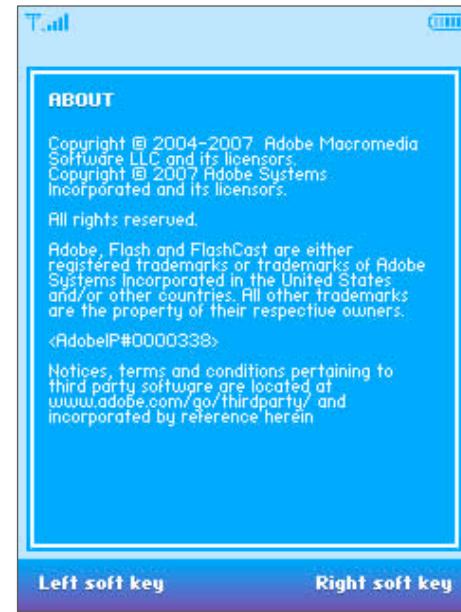
Also capable of displaying 'banner' information contained in individual channel feeds.

### Customisable views, dialogs and other elements used across the Flash Cast experience



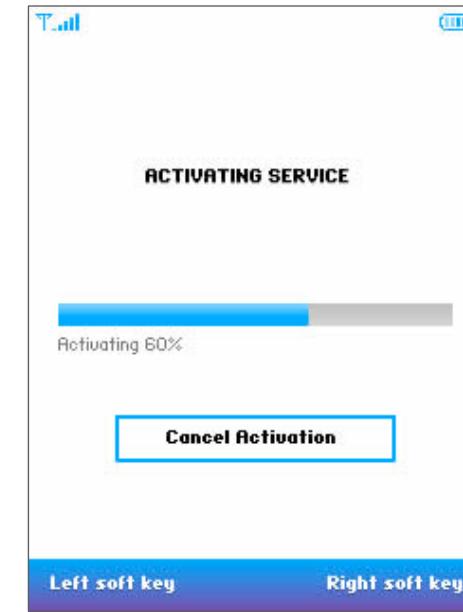
#### 9. Install progress (fc09)

Presented during channel downloading and activation.



#### 10. About (fc10)

Presented when 'About' is selected from the options menu. Normal contains copyright information or similar content.

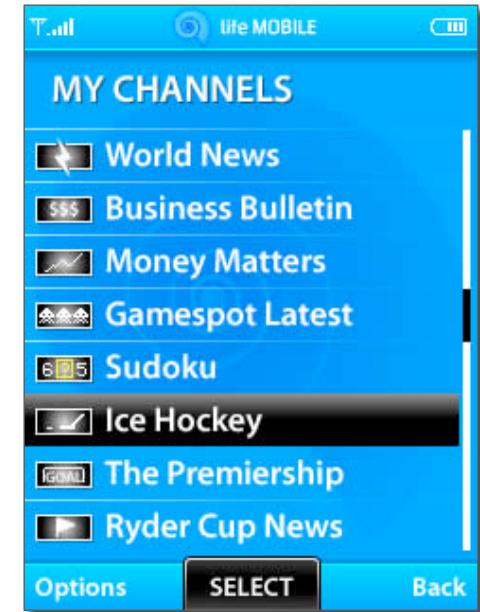
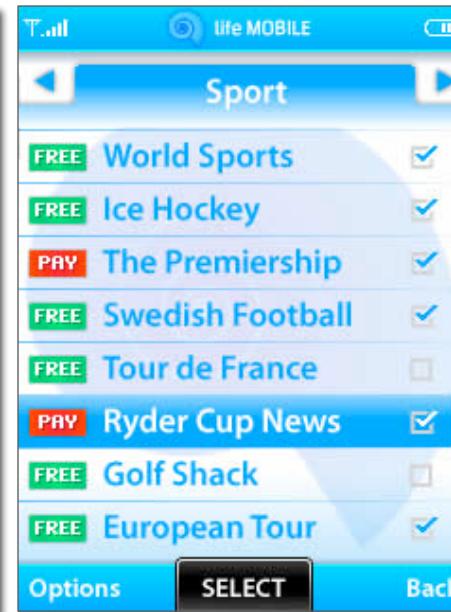
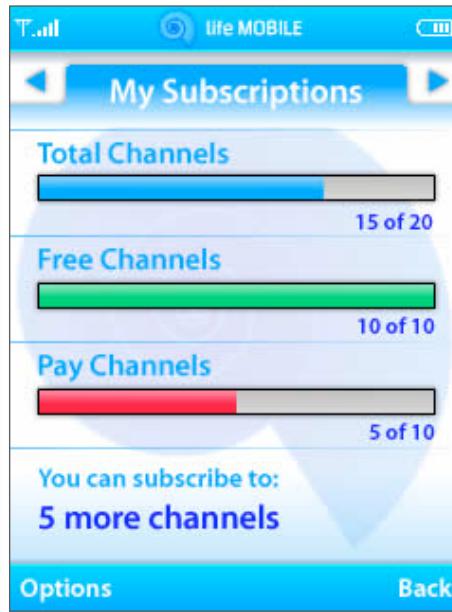
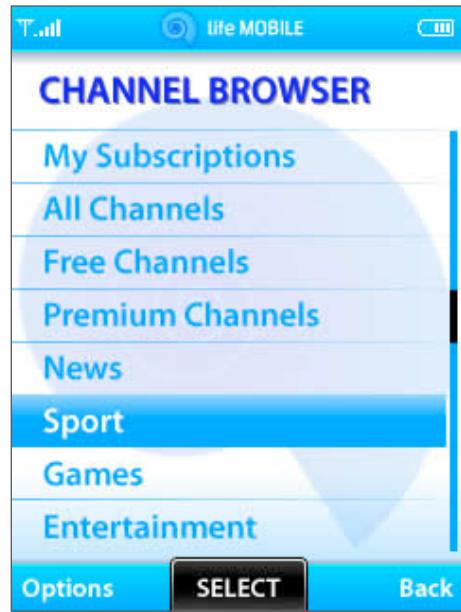


#### Init

Presented when system UI is being initially downloaded. Thus normally only displayed once or when a complete system refresh is required.

Very similar in appearance to fc09.

System UI can be extensively modified to suit the requirements of any deployment



### Customised Channel guide/browser (fc06)

Note custom softkey labels (fc01) providing support  
for optional centre softkey labelling.

### Customised now playing view

# Workflow tips

- Recommended tools
- Layout designs using Fireworks
- Fireworks symbols
- Importing Fireworks symbols
- Importing Fireworks .pngs
- Using Fireworks slices
- Tracing in Flash

**Fireworks and Flash are excellent partners for Flash Cast design and development**



**Adobe Fireworks CS3**

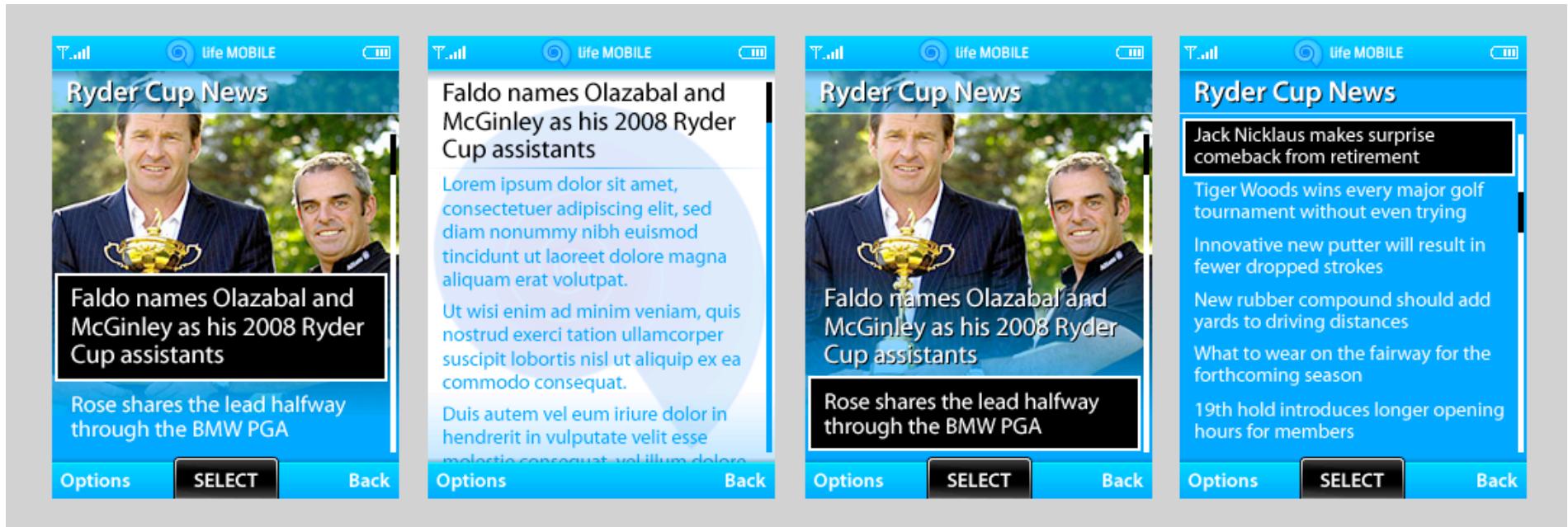
Optimised for screen graphics production



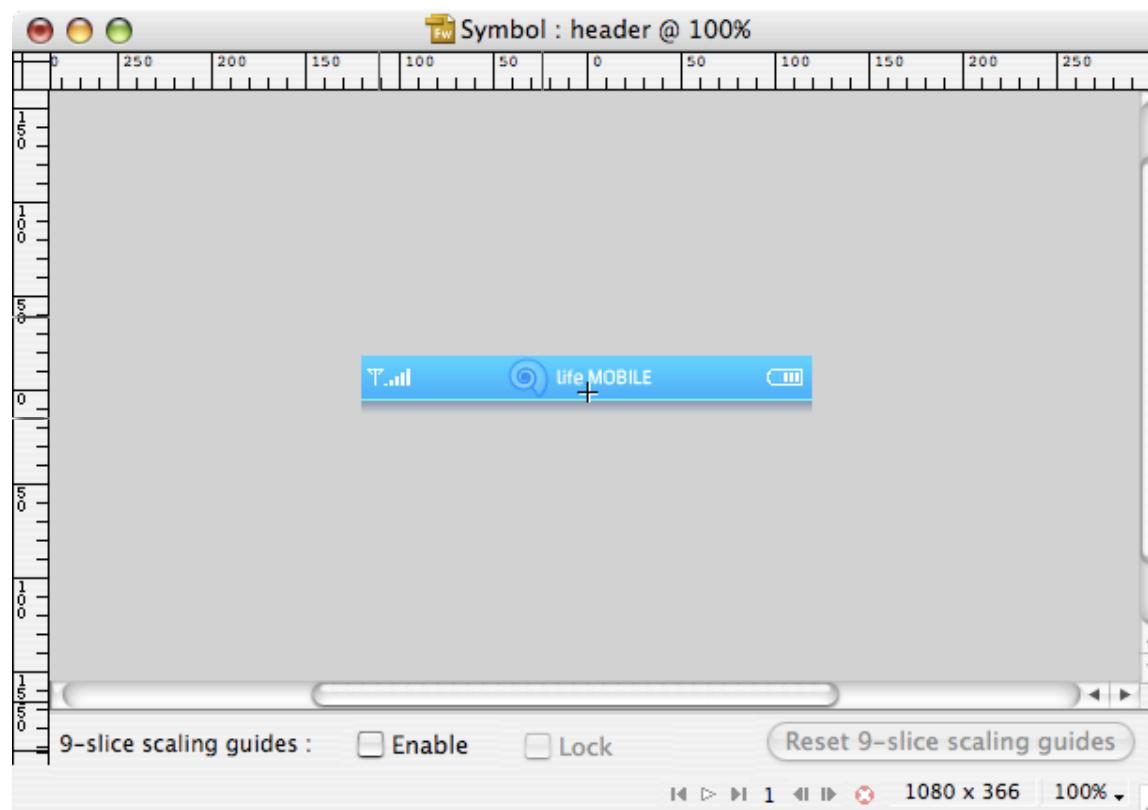
**Adobe Flash CS3**

The essential Flash Cast IDE

## Fireworks is an excellent rapid design tool



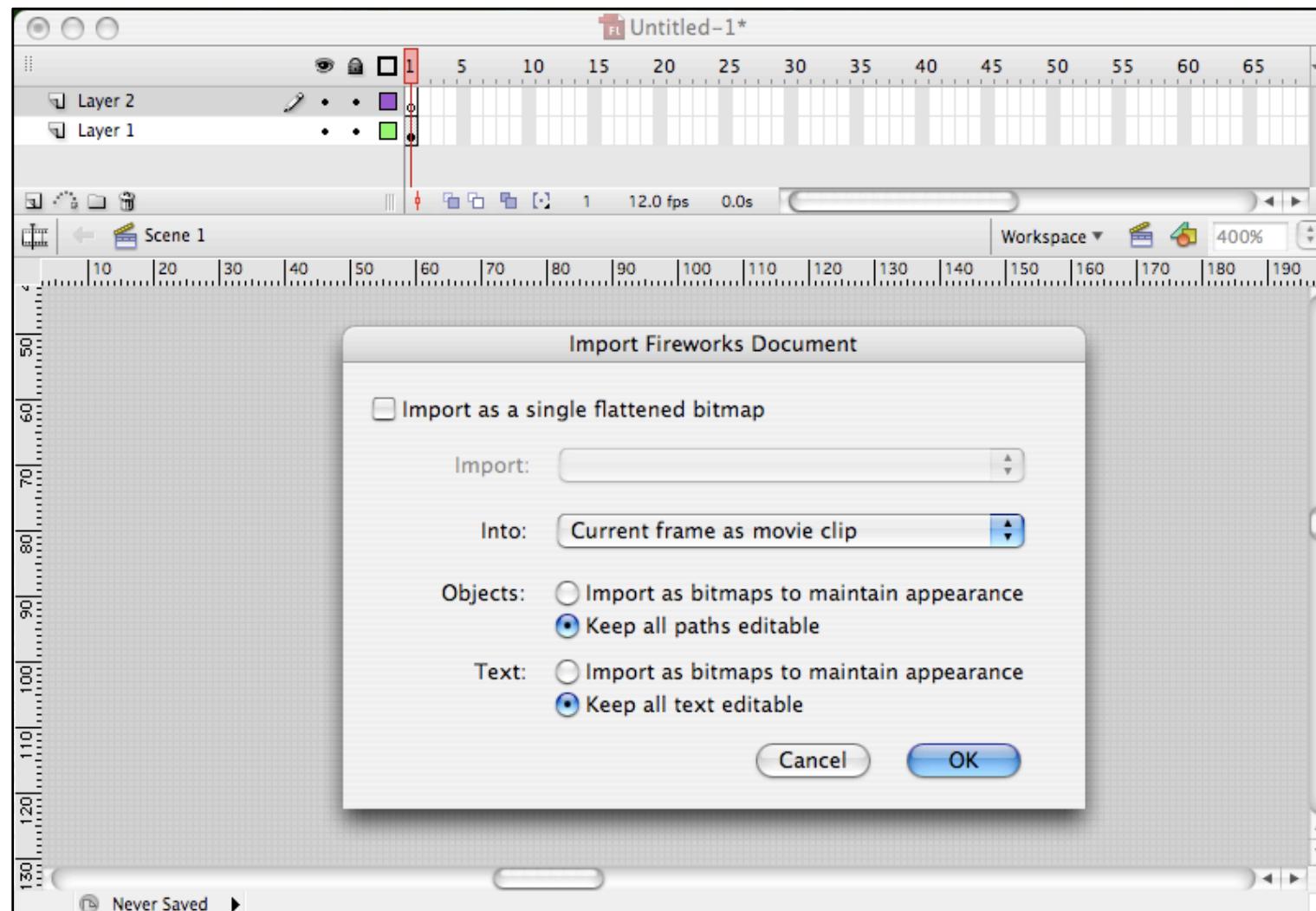
### Similar functionality to library items in Flash



#### Typical header symbol

Using symbols allows designs to be quickly assembled  
and rapidly changed when required.

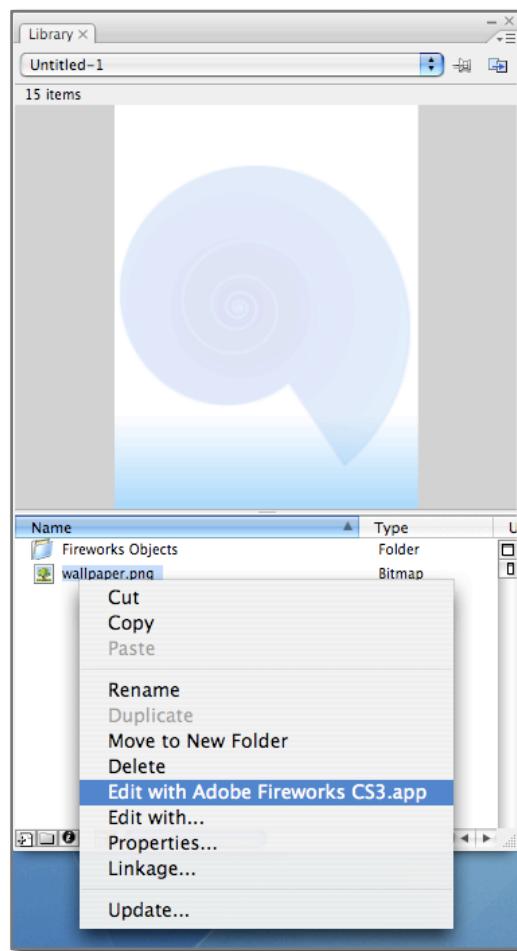
### Symbol structure can be preserved when importing into Flash



### Fireworks import dialog

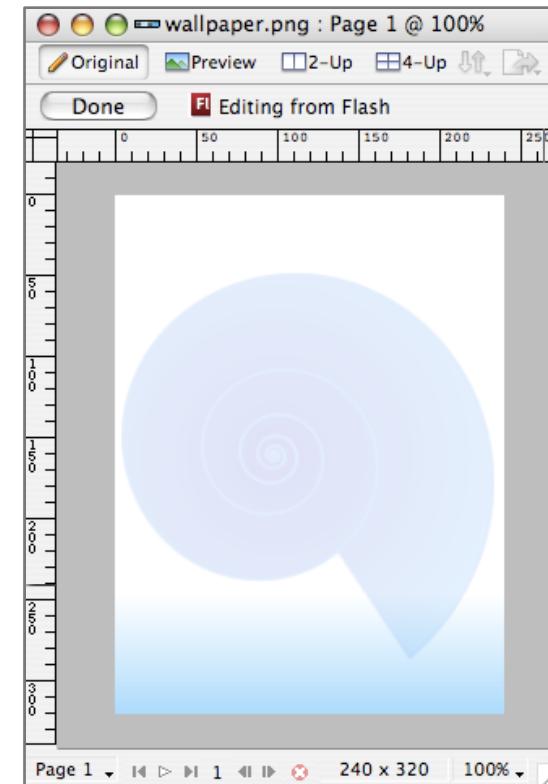
Maintaining paths and text makes imported symbols a good basis for the development of individual movieClips in Flash.

Bitmaps in the Fireworks .png format remain editable within Flash



### Contextual menu within Flash Library

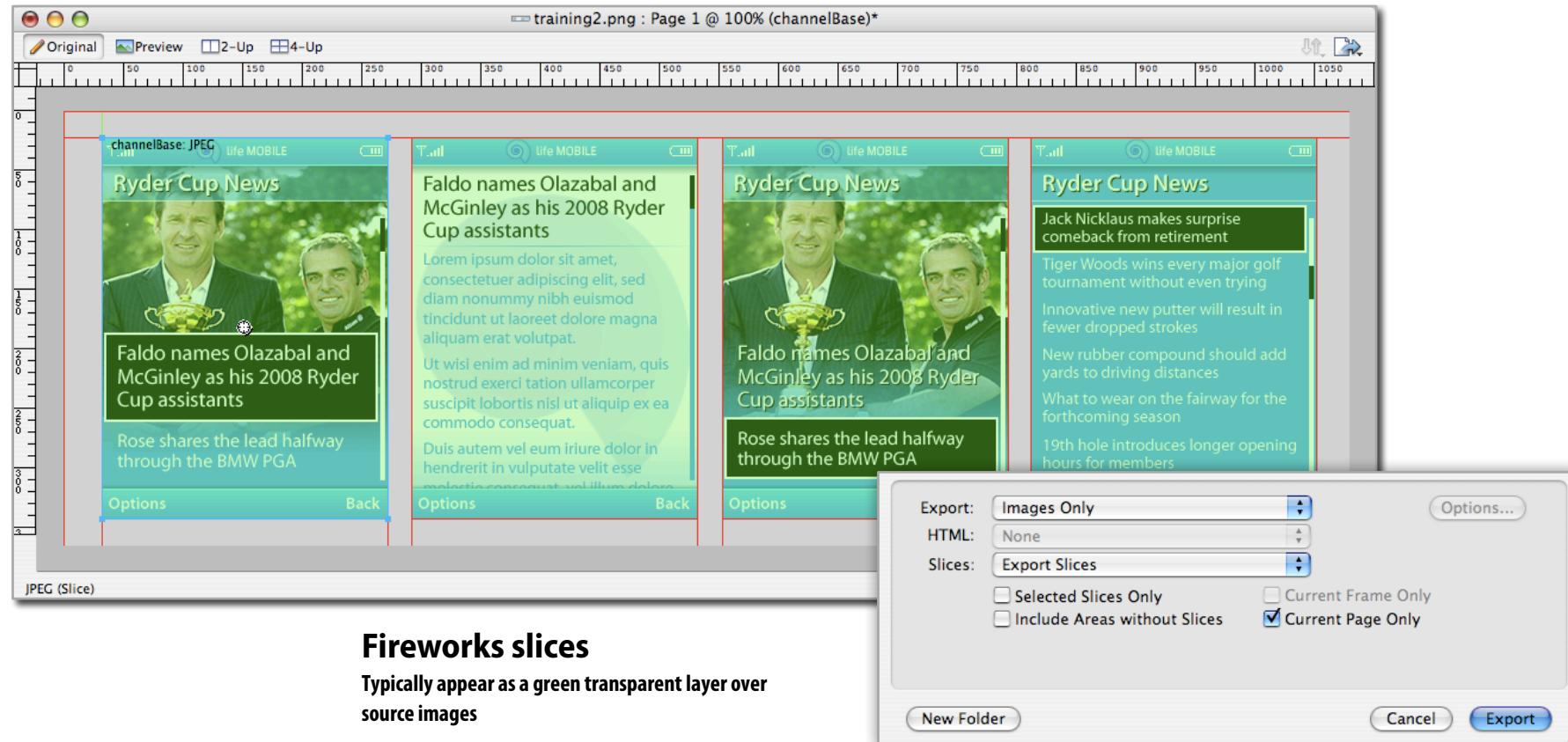
Imported Fireworks .png files can be edited from  
within Flash.



### Editing Fireworks .png file from Flash

Clicking on Done updates the .png file in the Flash  
library.

## Convenient method for exporting screenshots for use as trace images in Flash

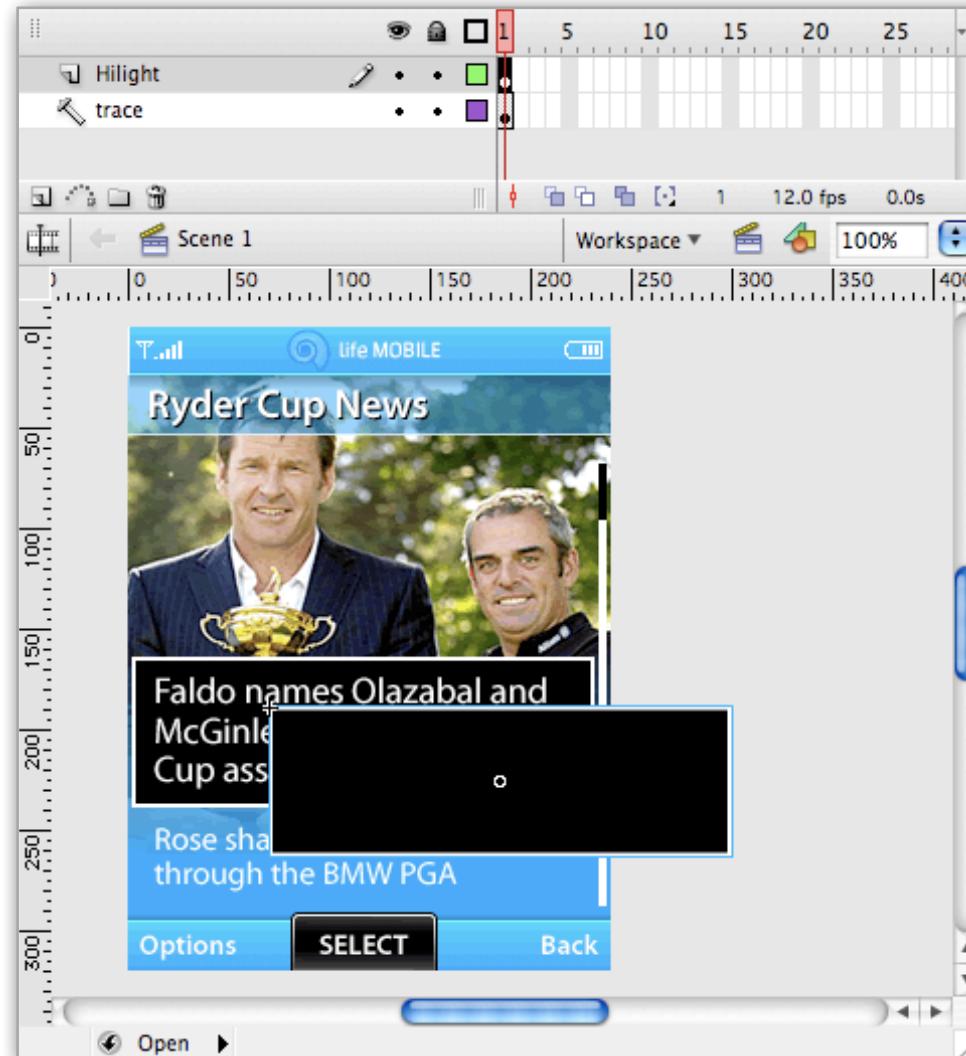


### Fireworks slices

Typically appear as a green transparent layer over source images

### Export dialog box

Exported Fireworks screenshots can be used for alignment and sizing



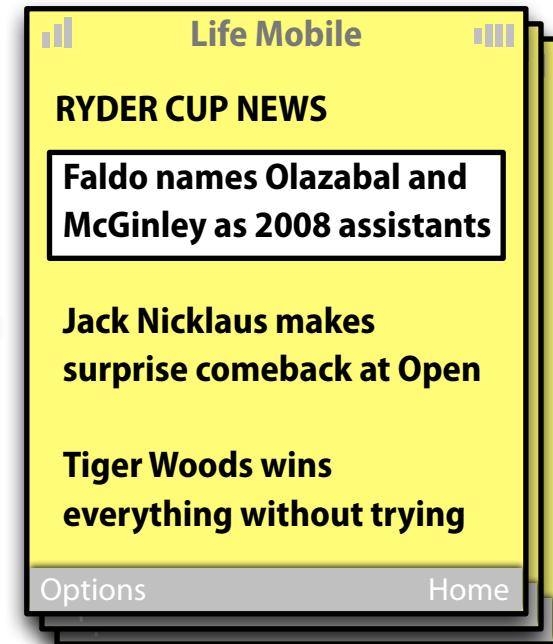
### Exported bitmap on trace layer

Make layers into guides by using the 'guide' function.

# Branding

- **4 main components**
- **Continuous branded experience**

Overall Flash Cast experience is dependent on 4 main components:



### Now Playing

Effectively the Flash Cast 'Home' screen the Now Playing screen provides the user with an easily navigable list of channels that they are currently subscribed to.

### Catalog

Provides a place for users to manage their subscriptions and browse content by category. Can contain multiple screens.

### Individual Channels

Communicate the actual content to the user.



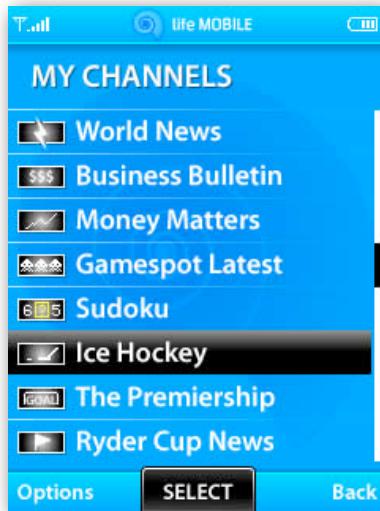
### Other system UI

Options

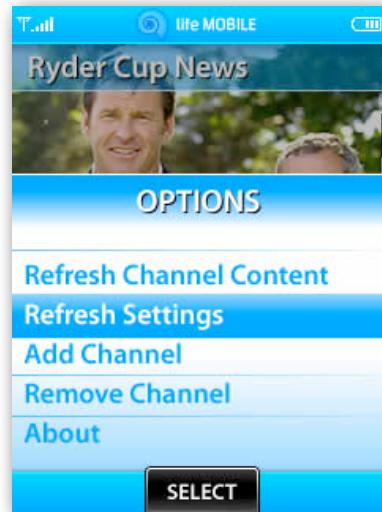
Exit

Includes standard system items such as soft keys, menus and dialogs

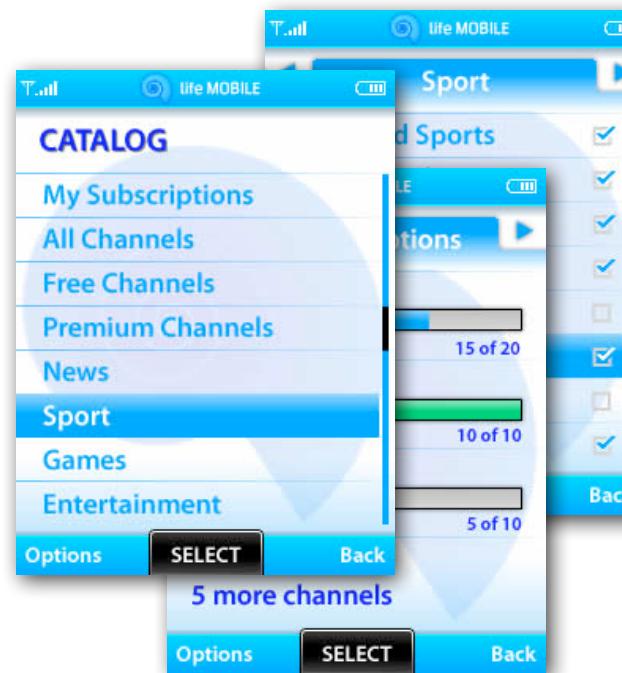
Consistently brand these 4 to create a continuous branded experience



**Now Playing**



**Other system UI**



**Catalog**



**Individual Channels**

<End of presentation>