

Developing Dynamic Content for uiOne™ Delivery System

David Berry, Hani Naguib
QUALCOMM Cambridge



Presentation Overview

- **Trigs and Triglets**
- **uiOne™ Delivery System**
- **uiOne™ Publisher**
- **Triglet Workflow**
- **Extending uiOne™ Publisher**
- **Feeds**
- **Developing an RSS Feed**
- **Demo**
- **Content Server Adaptors**
- **Q&A**

Trigs and Triglets

- **Trig**
 - Updateable User Interface

- **Triglet**
 - UI Update
 - Collection of Resources

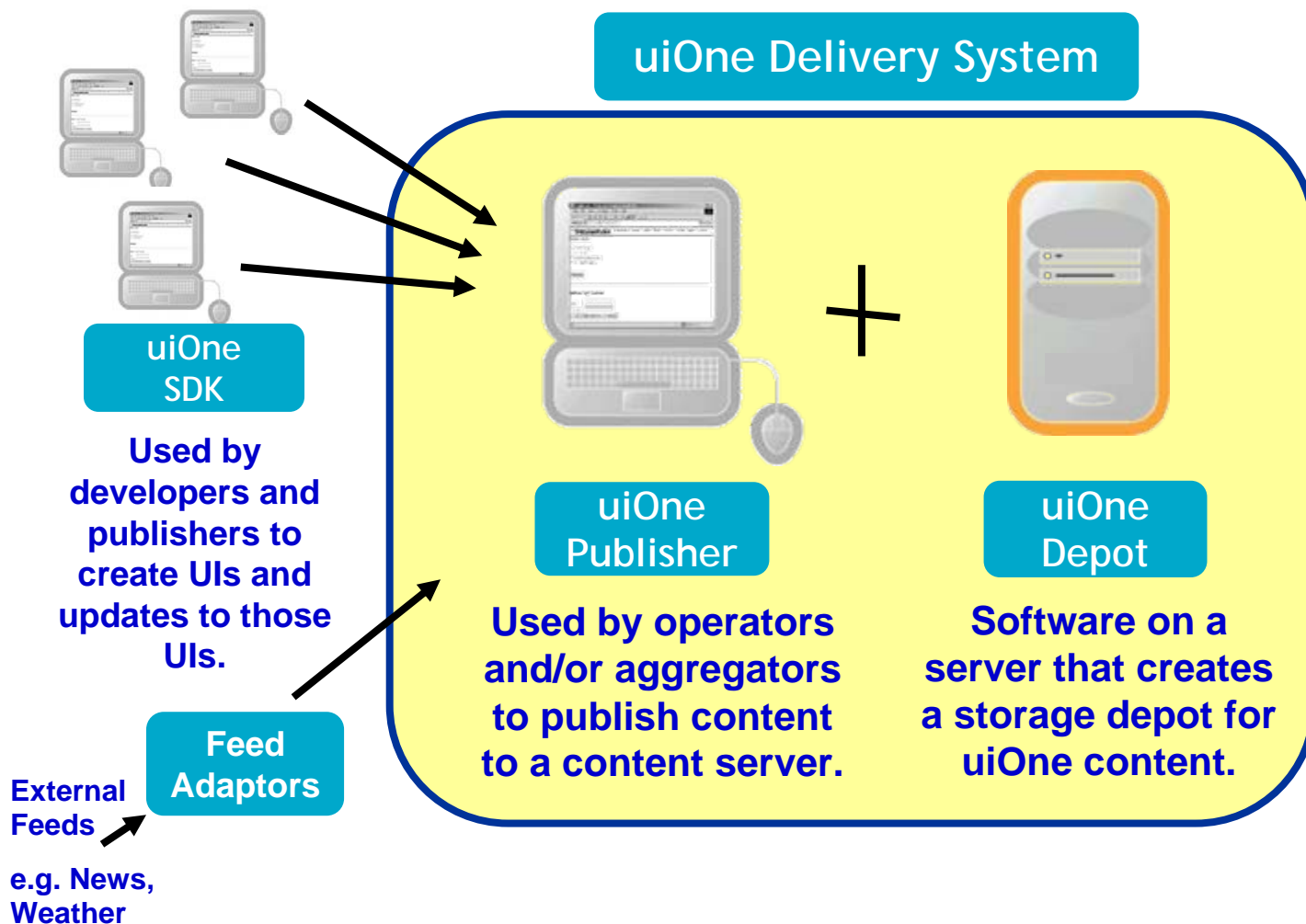


text/0/temp	10°C
Image/0/icon	
text/0/description	Partly Cloudy

Triglet Usage

- **Maintenance**
- **Promotions**
- **Menus**
- **News feeds – typically periodic**
 - Weather
 - Cartoons, Wallpaper
 - Stocks
 - Sports scores, league tables
 - Event timetables

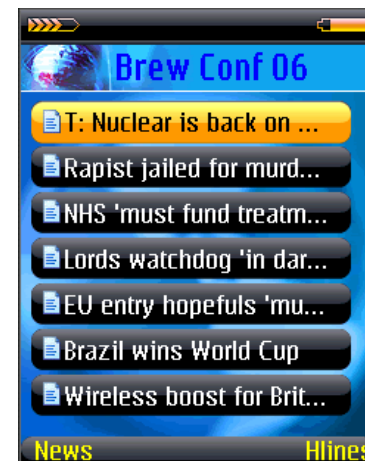
uiOne™ Delivery System Overview



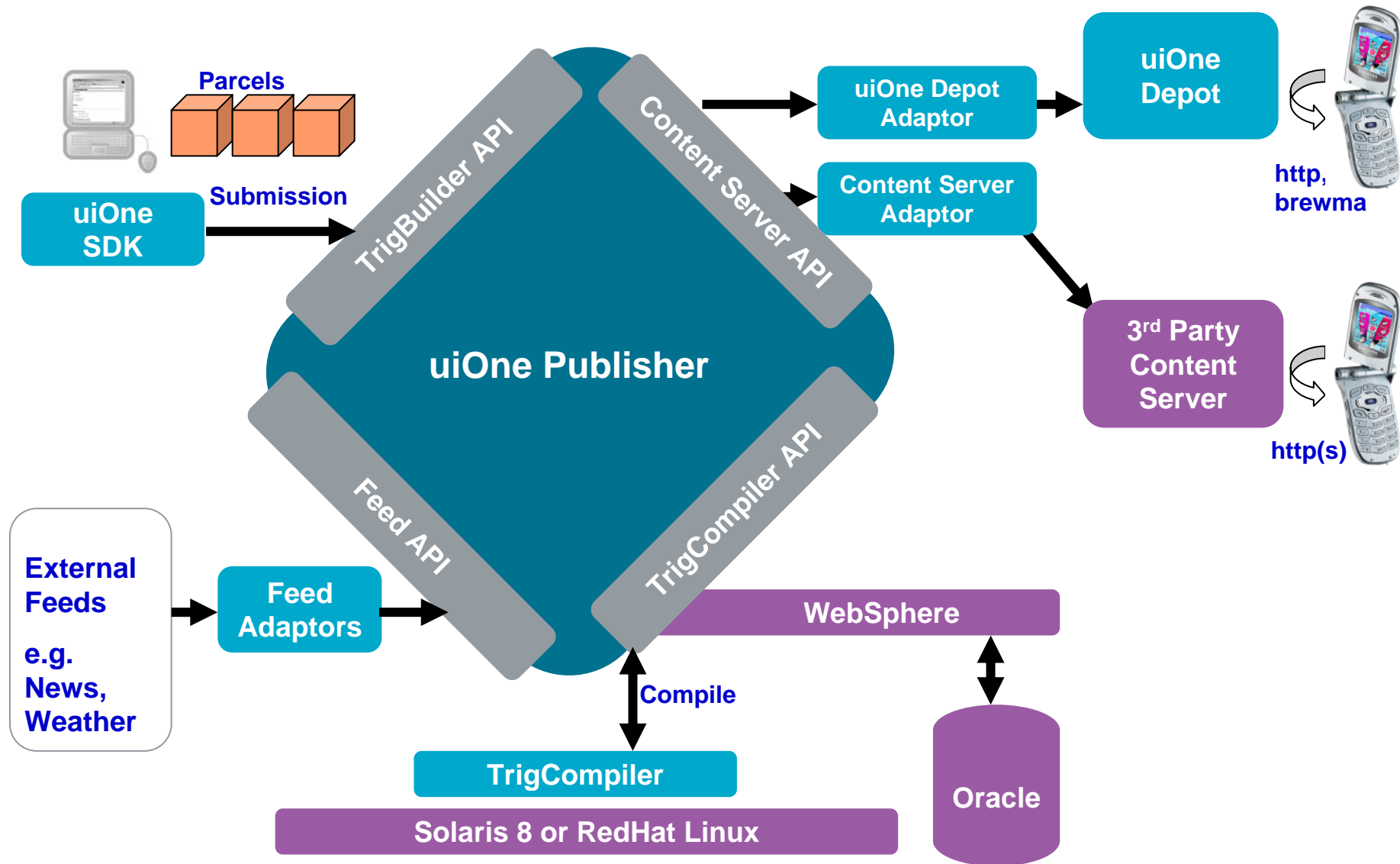
Updates this



to this



uiOne™ Publisher Architecture



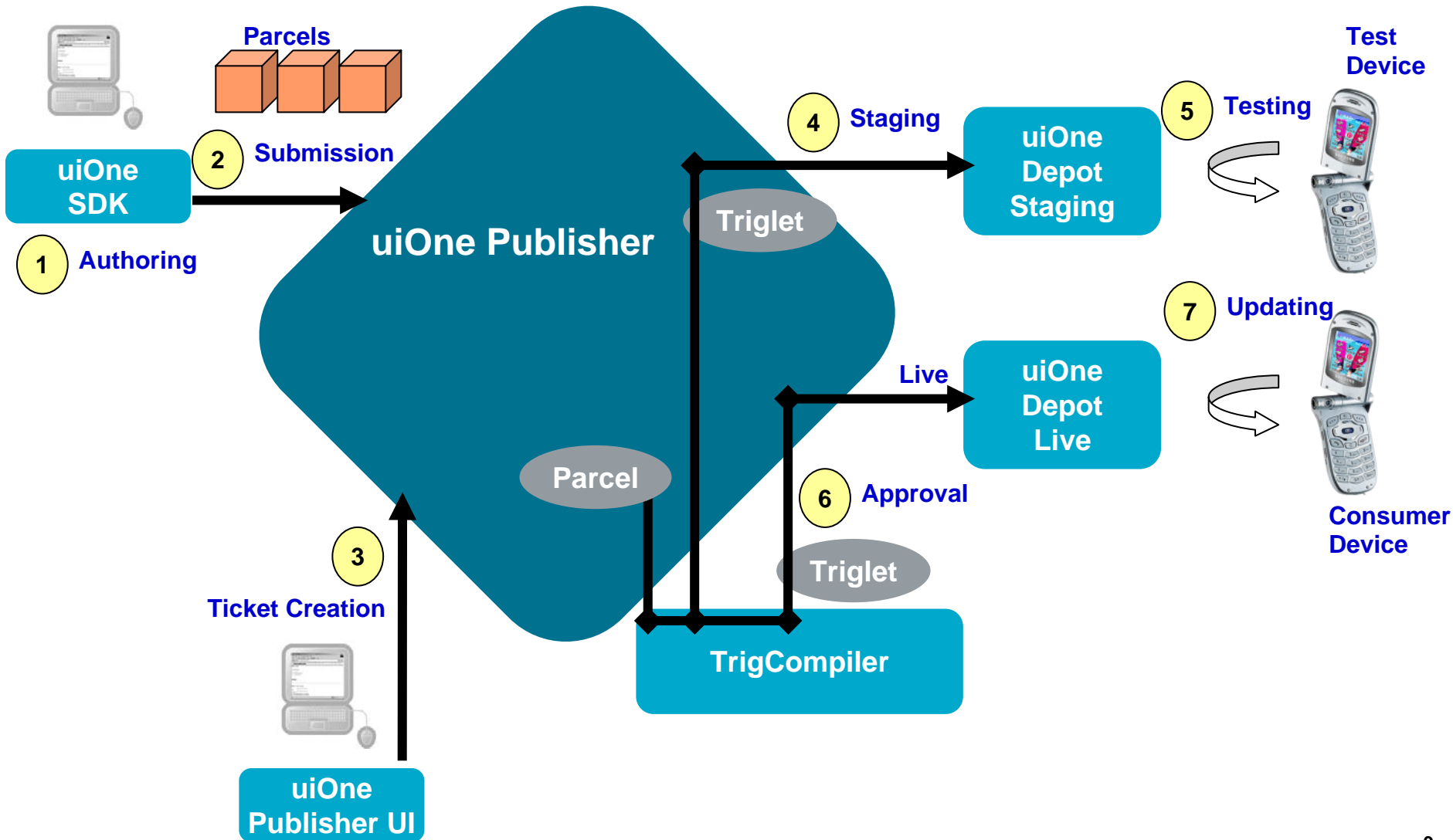
uiOne™ Publisher Functionality

- Management and publication of partial UI updates (Triglets) for BREW uiOne-enabled mobile devices
 - **Stored in a device and language independent format**
- Management of compilation of Trig(let)s for individual languages and device types
- Upload of compiled Trig(let)s to content servers such as uiOne Depot

uiOne™ Publisher Functionality

- uiOne Publisher allows staged deployment of Trig(let)s
- Automated retrieval of content from external data sources and repackaging of this content in Triglets
- Adaptor SDK for extending uiOne Publisher

Triglet Workflow

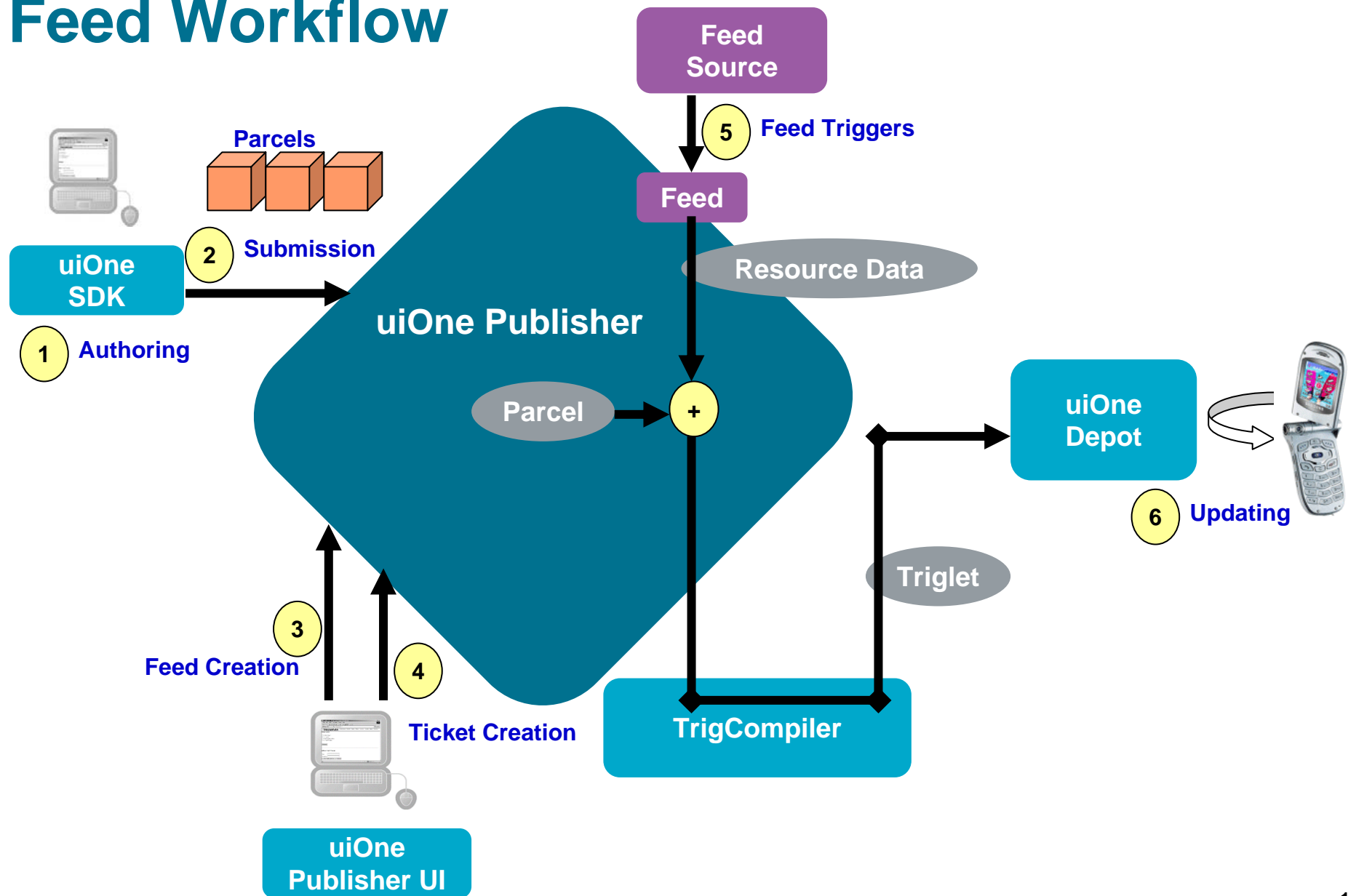


Triglet Workflow Demonstration

Feed Adaptors

- **Feed Adaptors extend uiOne Publisher**
 - Provide a mechanism for injecting dynamic data
 - Convert data from source form to Triglet form
 - Periodic
- **uiOne Publisher provides two built in feeds**
 - RSS
 - Clock feed used mainly for testing

Feed Workflow



uiOne Publisher Adaptor Developer SDK

- **Supplied with uiOne Publisher release CD**
- **Contains:**
 - Adaptor SDK jars
 - Sample Adaptor Code
 - Feed Adaptor
 - Content Server Adaptor
 - Developers Guide
 - Java Doc

FeedAdaptor Interface

Feed Adaptors must implement the FeedAdaptor interface

```
public interface FeedAdaptor {  
    Version getVersion();  
    FeedDescriptor getDefaultDescriptor();  
    void validateDescriptor(FeedDescriptor desc);  
    void run(FeedDescriptor desc, FeedUpdater updt);  
    void stop();  
}
```

Java Code Demonstration

Feed Creation User Interface

Config
Staging Area
Live Area

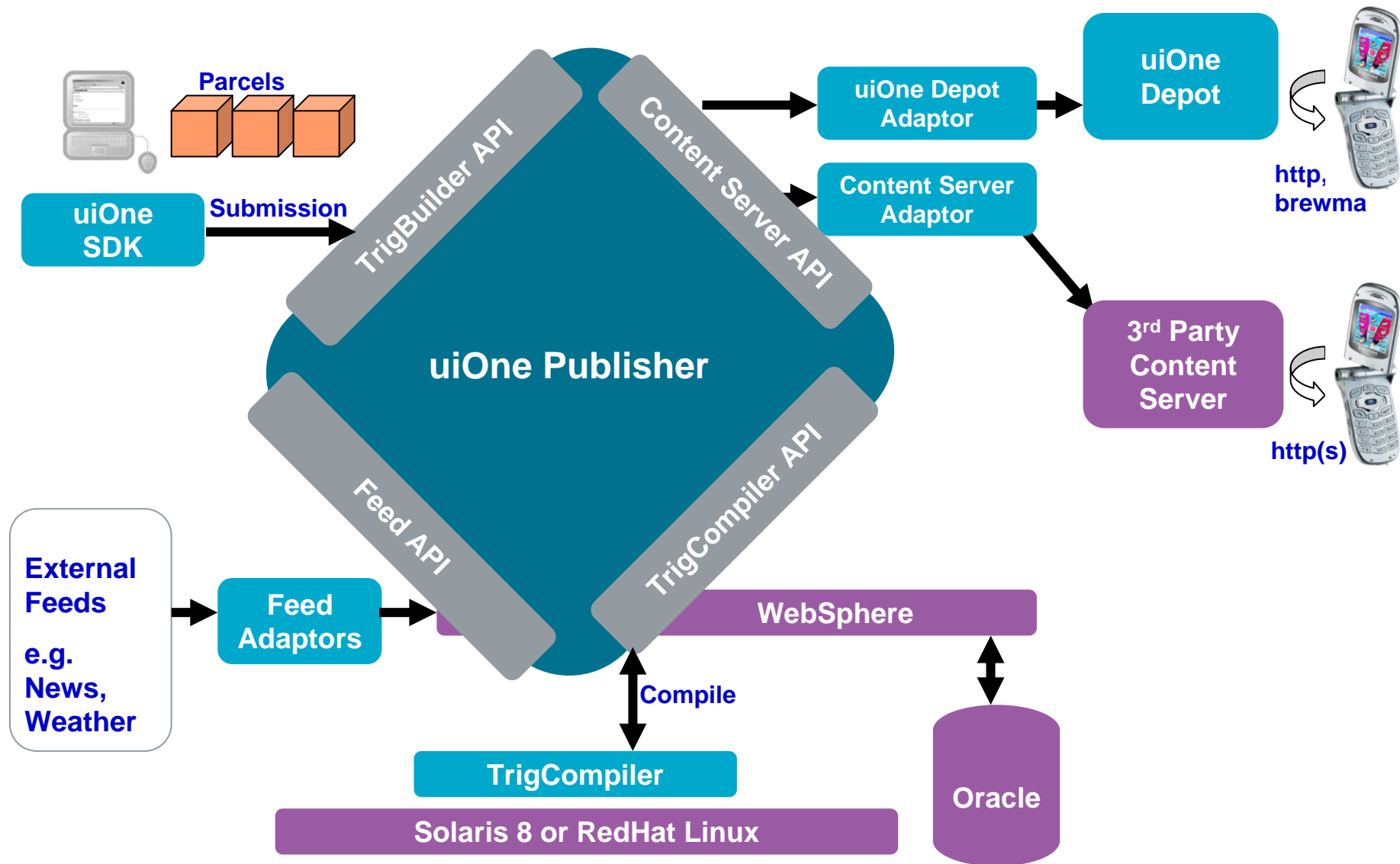
Feed Settings

Feed Name	<input type="text" value="BrewCon RSS Feed"/>
Feed Description	<div style="border: 1px solid gray; height: 100px;"></div>
Feed Adaptor	BrewCon RSS Adaptor
Feed Settings	
URL	<input type="text" value="http://newsrss.bbc.co.uk/"/> The url of the rss feed.
Period	<input type="text" value="5"/> The periodicity of this feed in minutes

Feed Resources	
Title	<input type="text" value="news/%d/title"/> The RSS item title.
Description	<input type="text" value="news/%d/summary"/> The RSS item description.
Link	<input type="text" value="news/%d/link"/> The RSS item link.

Feed Workflow Demonstration

uiOne™ Publisher Architecture



Content Server Adaptors

- **Content Server Adaptors extend uiOne Publisher**
 - Provides a mechanism to make the content available for distribution
 - Support for 3rd Party Content Delivery Systems
- **uiOne Publisher provides two built in Content Server Adaptors**
 - Local File
 - FTP

ContentServerAdaptor Interface

Content Server Adaptors must implement the ContentServerAdaptor interface

```
public interface ContentServerAdaptor {  
    Version getVersion();  
    FeedDescriptor getDefaultDescriptor();  
    void validateDescriptor(FeedDescriptor desc);  
    boolean dispatch(ContentServerDescriptor desc,  
                    ContentDescriptor content,  
                    byte[] contentData);  
    String getDeviceURL(ContentServerDescriptor desc, ContentDescriptor content);  
    String getWebURL(ContentServerDescriptor desc, ContentDescriptor content);  
    byte[] getContent(ContentServerDescriptor desc, ContentDescriptor content);  
    void stop();  
}
```

Q & A