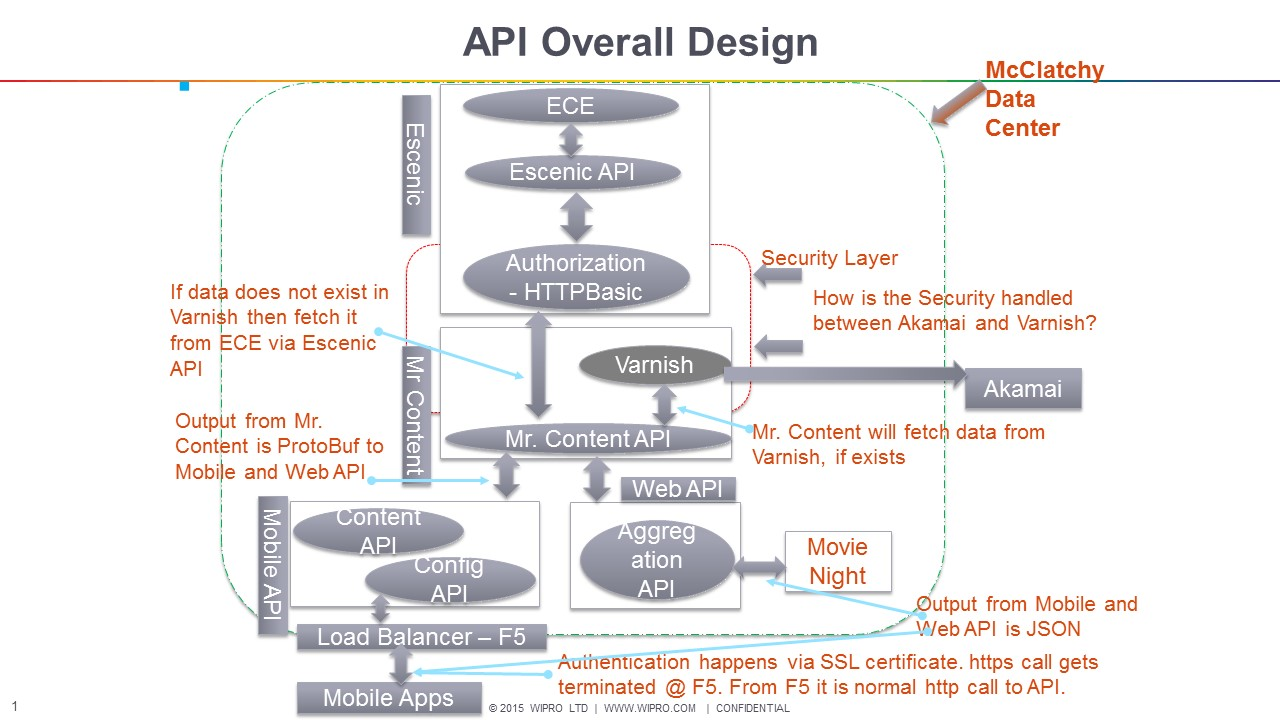
**API**

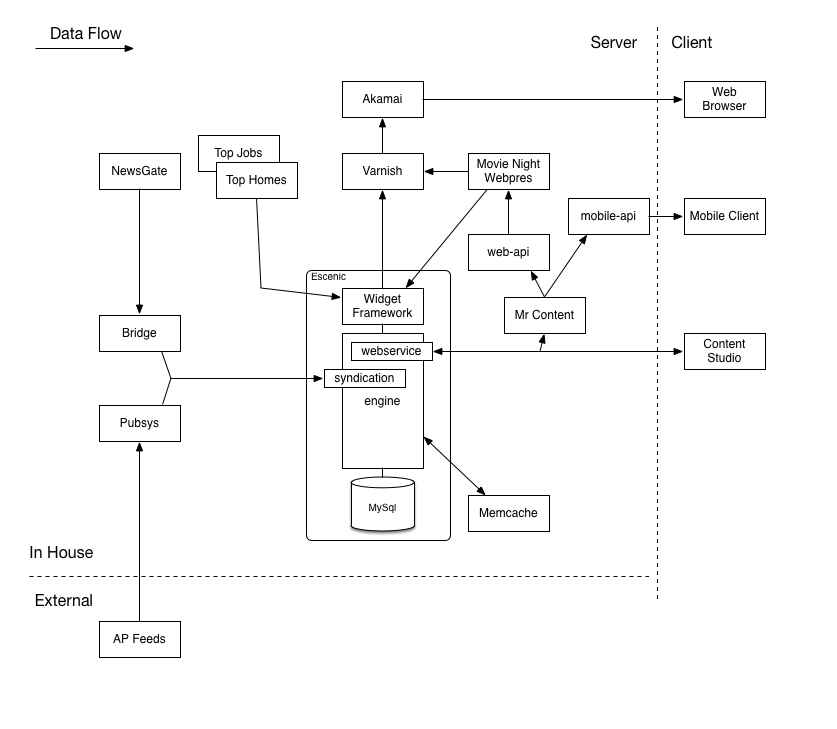
****

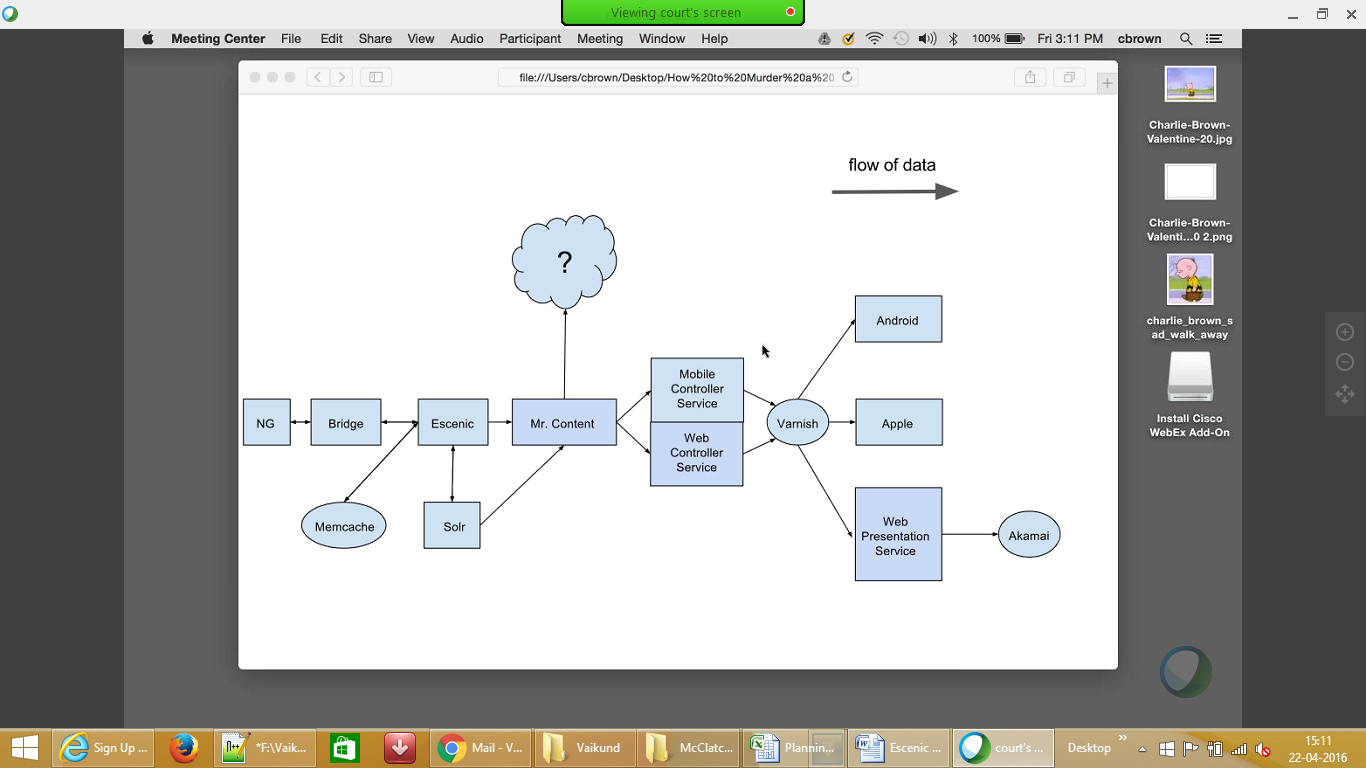
**API**

* Mr. Content APIs used by Web and Mobile
* Service Repo not actually used
* Session on Architecture of Mr. Content, Mobile API/Config API, Web API for Movie Night
* Mr. Content architecture documentation to be done
* Code walk through - one dedicated session
* Mr. Content has API and Cache
* Mobile is segragated into Content and Config APIs
* Web API serves Movie Night project
* Security layer between Mobile APPS and Mobile API is done via SSL
* Security layer between Mr. Content and Escenic is via username and password... authentication done via HTTPBasic
* 3rd Party content is populated via syndication into PbuSys and then propogated to Escenic
* Agregation API is for Video for Movie Night
* Testing APIs - done using Postman, development team does not have bench mark for response time.... testing team does have...
* Health check - xymon
* Regression test suites executed daily by cron job
* Logger done using Supervisor
* PubSys (Internal CMS/Aurora) is having 3rd party APIs
* Varnish Layer (Reverse Proxy & Caching) – caching static pages  & Memcached – caching Key Value pairs
* HTTPS – F5 – SSL Termination – SSL Certificates
* Protobox requirement
* API Life Cycle in McC
  + Escenic native API development & customization based upon user stories
  + Go wrapper API development - deployment is done thru net/http
* Etag, Cache Control, Keep Alive & other Request & Response Headers
* Feed & Syndication process
* Alok’s queries on Scalability & HA will be discussed during DevOps planning session

**Escenic & NG**







**NG & Escenic**

News Access used to create the folders for the journalist to upload the content, images and videos

News Director - used by the journalist to create stories (includes text, images and videos)

Stories (text, images and videos) are uploaded into the folder

Same folder can contain different stories for different publication

Workflows can be bypassed (journalist can directly send the story to Escenic) by directly setting the status to Send to Online in News Access

Via News Access a story can be set to journalist and he/she can work and update the story and via Email send the story back to News Access

Once when the story status is set to Send to Online, content (text, images and videos) will be zipped along with 2 xml format (one with .org extension and other .xml extension).

.org file is just for audit, Escenic uses only the .xml.

.xml contains all the information about the story (including all the metdata of the images and videos)

Creating zip files with all the related information about a story is done via bridge (bridge - CCI's plugin which transfers stories from NG to Escfenic. It is 2 way - changes done on stories in Escenic can be propogated back to NG - Currently McClatchy does not prefer to send the updated story back to NG). Eventhough Bridge is provided by CCI - McClatchy does make changes according to their business needs. Bridge is developed in Java

WireFeed (also called as Auto Feeds) come from 3rd party news agencies (like AP and Reuters) - These first come to PubSys (Aurora - internal CMS) - from PubSys - a Syndication process updates Top homes and Top Jobs DBs - Need to have a session on this

Other type of feed is from MNN (McClatchy National News) - This feed is from NG - This is data is exported into Escenic CE via the bridge

Via Videolicious - journalist upload vidoes - authentication done via McClatchy - username password - Valid users can save their videos into a FTP server.

All the applications are devloped using Escenic Widget Framework (WF)

Wijita site is not part of Print - still the story is developed in NG but shared only for Digital

WF directly pulls the content from ECE

Due to performance issues in WF - a new project named Movie Night is being developed - First release has happened in March - all the Sites video section is populated using the templates developed in movie night

project (Movie night - NodeJs, Nunjucks and Express). Plan is to move from WF to Movie Night

For performance – Escenic also uses Memcache – but this is maintained by Escenic – McClatchy does not do any maintenance on this…. Stores information of the story as an object (key is created using object id and version id). Need to have KAP session on how the cache is being rebuilt when the servers are brought down and back up.

WF - for video sections calls template developed in Nunjucks as they are stored as individual HTML files

Pages are cached in Varnish and Akamai - Akamai is the superset and apps fetch information from Akamai 80% and from Varnish 20%. Akamai is the CDN - contains both text and images. 2 Types of caching in Varnish – In-Memory and Global. WF first checks in In-Memory and then Global Memory and then if the data does not exists in both then fetches it from ECE.

Escenic relies heavily on Varnish. - Need a session on how data is stored and fetched from Varnish, how data is stored in Akamai and how Varnish transfers data into Akamai

Architecture - Brian to share the design (how the servers are linked and the data flow between different components)

Related to Advertisement - we could not cover so going to have a full session during KAP session

Apache Solr is used for search - Full text index search is not implemented - search currently on metadata tags - Session on how the Solr REST Api called from WF

Solr engine is running on 18 servers inside their DC

During DevOps session need to check on Solr implementation (how many servers, load balancing, is it HA)

Snapshot - migrate configuration from one publication to another

JMS Montiroing used to check the health of all the hosts - running on Tomcat

Need to have a session on schedule jobs during KAP - Code walk through

There are only 4 jobs that

Stories from NG imported into Escenic CE - in Escenic format (format and structure - we need to havea session during KAP)

Escenic Content Studio - Java based desktop application used for making changes to the stories from NG

DCs in Miami and Fort Worth - 39 servers each - more info from DevOps

DR not completely setup - more info from DevOps

**Escenic In-Flight Projects**

Movie night is the main focus, complete implementation of video list, video content display

So far, video list and video content display is live in movie night

Replacment of header & footer in Widget framework are current focus in movie night

Escenic Widget framework JIRA tasks & bug fixing is also in-scope

Movie Night deleivery time lines are not identified/confirmed so far

Movie Night UI will replace Widget framework in future

Katie's team is working on the Movie Night UI implementation tasks

Paul D's Team is working on the Wire feed & Bridge enhancements. They are also working on Video related changes.

Buy a photo featute implementation also planned for future release

PubSys CMS system is the legacy CMS systems develeped using PERL script by McClatchy team (By Walter & Team)

Seperate deep dive session planned for Wire Feeds /JSON feeds (extends RSS feeds)

**Testing Framework for Escenic Development**

Java VisualVM framework build using Ruby - Cucumber is used for UI functional automation

From Cucumber script Java class is called for regression testing

In futute we may need to extend testing focus on NewsGate API which NewsGate is currenlty working on

Validation of images and videos are manual process. Text validation done by the Editors before coming into Escenic CE.

Test data prepration & manintenace - Hari to fill on this

Test environment ownership, deployment & co-ordination  - Hari to fill on this

QA/Testing entry and Exit criteria

QA Gating sign-off process

Jenkins test jobs  configuation & related DevOps tools

**Movie Night**

Goal of Movie Night implementation ->

To replace Widget framework with Open source systems, HTML & JavaScript templates

To reduce overheads and time consumption in UI developement & maintenance

Escenic application runs on Tomcat server and Movie Night on Node JS.

Movie night implememtation started with Video and video list.

Varnish caching may be used for Movie Night Web presentation implementation when the whole site goes live on Movie Night.

High availabilty of the application and load balacing is focused ( 3 to 5 server instances) in dockers.

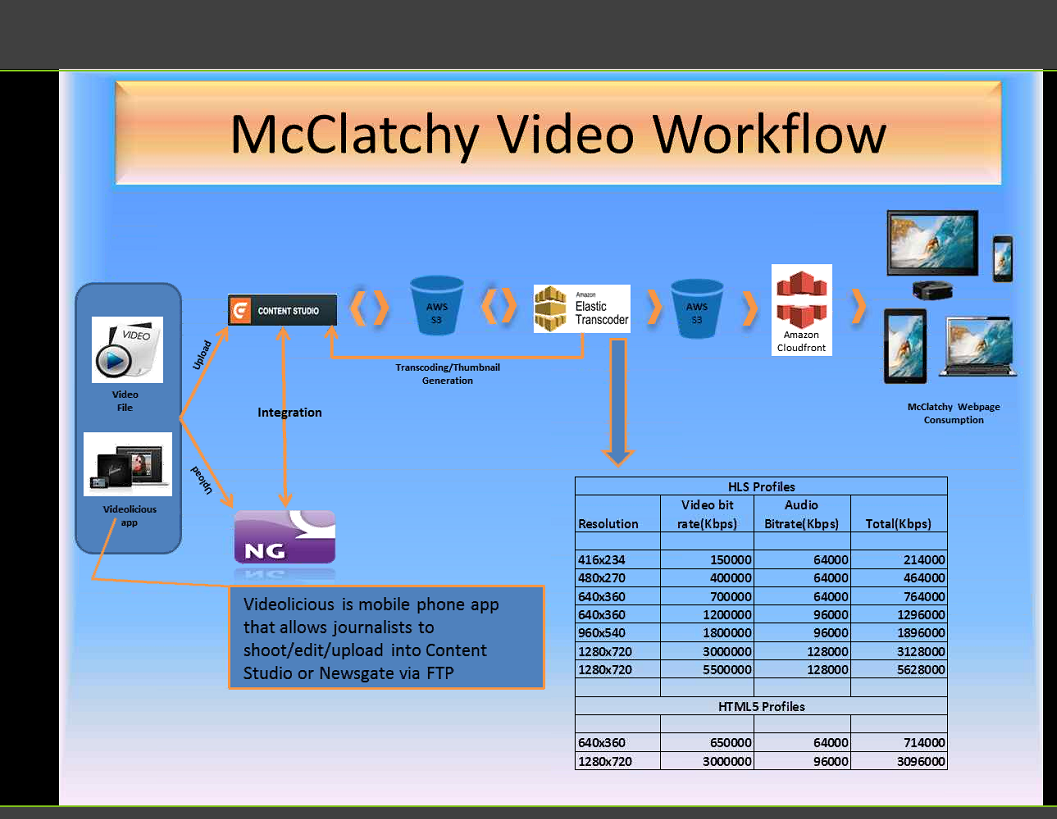
Dockers is part of DevOps (plan for separate session part of DevOps).

Mokawk framework is used for unit testing (to test the Node JS scripts).

Moview Night application Releases will happen for all sites together.

Layout config both Escenic and Movie Night use the same feature file from Escenic

**Video**



* 2 ways of video
* Video Wish list
* Capture video and edit Video
* Upload from content studio - Essenic Content studio
* Video uploaded via FTP
* Assigning to Heading/Tag/session to display the video
* store in Draft - to hold the video not to be displayed
* Video upload to NG – Testing in progress, will be release ASAP
* Video moves to AWS – to have preview then publishing phase
* HLS profile – for Mobile device
* Videolicious overview for mobile apps
* MP4 for desk tops
* HLS profile – have created a resolution / rate/ bitrate
* Subtitle – have a road map in future
* S3 – Storage and changing how is managed – need to discuss with architect team
* JW players/ 360 Video
* Story are split into different content tickets in Jira
* Ad blocker and ad blocker replacement approach
* Content with third are included to display
* Metric and volume of content tracked for future analysis
* Skipping/View/Accessing from the end user information
* Video specify work with local ad team work to address this ad’s
* Testing on profile is not available, basic testing…

---------------------------------------------------------------------------------------------------------

* Upgrade based on release note and address the changes for the new version
* Changes handed on new version or browser change (Safari/Chrome)
* Video –> Escenic –> Amazon cloud – Handled on MP4, HLS
* An overview on Amazon cloud set-up and used by McClatchy
* Error handling – failed to upload or corrupt, no clear verification done
* SRGB – color validation
* About bookmark for video plays
* Upload process – JWS player send request via Web API and store data
* Standard Player style for each market, very basis color value, no change in control just modify the skin of the video
* Video Cashing
* Authorized done using profile – authentication – Add/Edit/Modify based on role
* Error handling used for video wish list
* MRSS feed for paid service in future

---------------------------------------------------------------------------------------------------------

**Testing on Video** –

1. Adjust video
2. Data from Web API for testing
3. Test for Ad tag’s
4. Play list, multiple video
5. Escenic uploading
6. No specify testing plan, just generic testing
7. No QA resource in video team,
8. NFR – no requirement – Beach mark, format, quality
9. Automation tool – mocha <https://mochajs.org/> - JavaScript for API testing
10. Testing point of contact for video testing - Scott, Phillip

---------------------------------------------------------------------------------------------------------

**Inflight on Video**

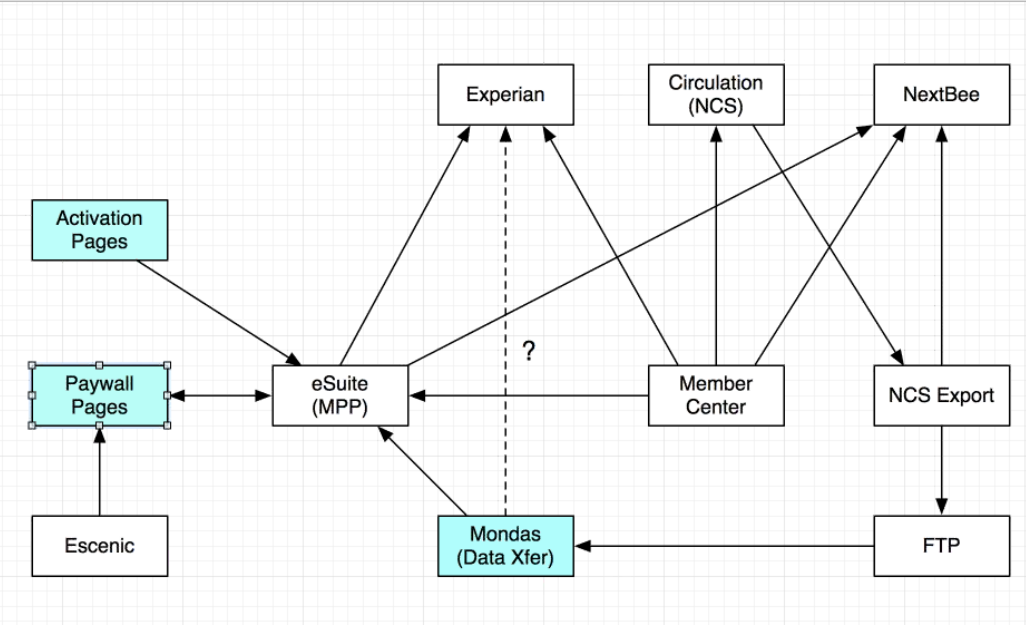


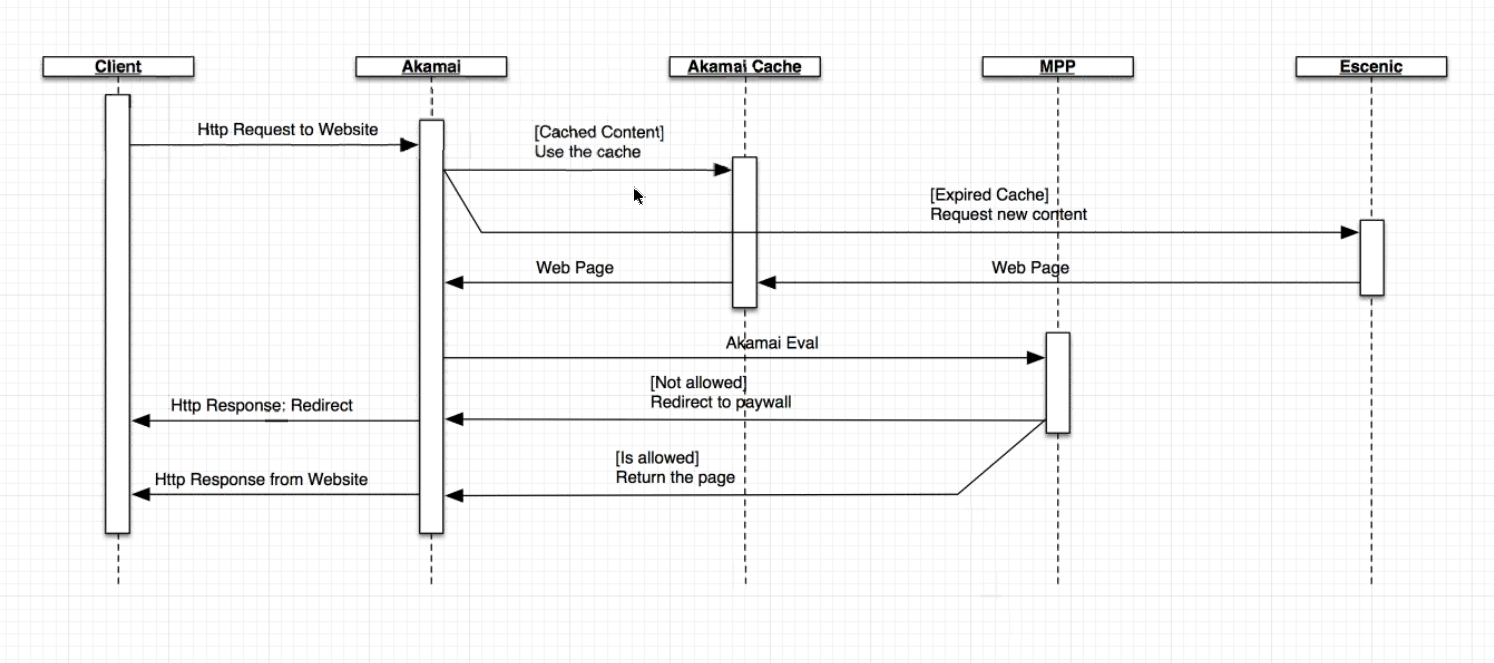


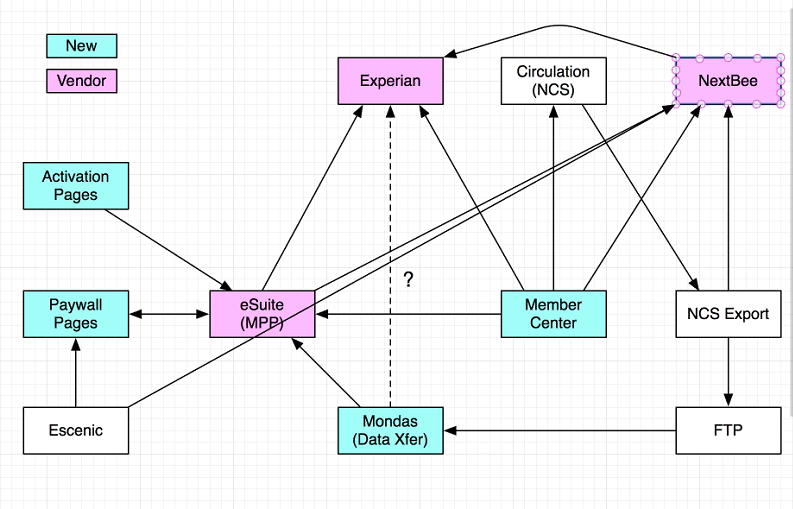
**Audience**

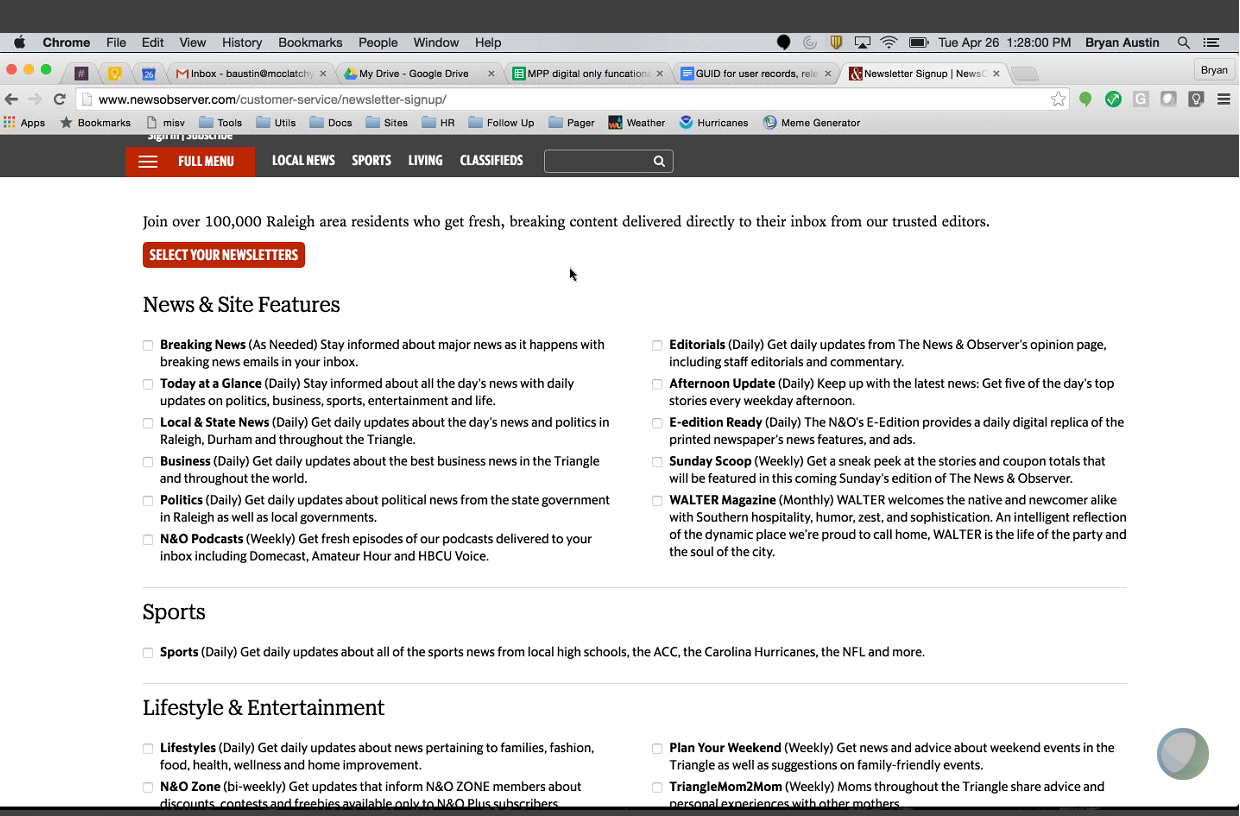
**Audience Notes:**

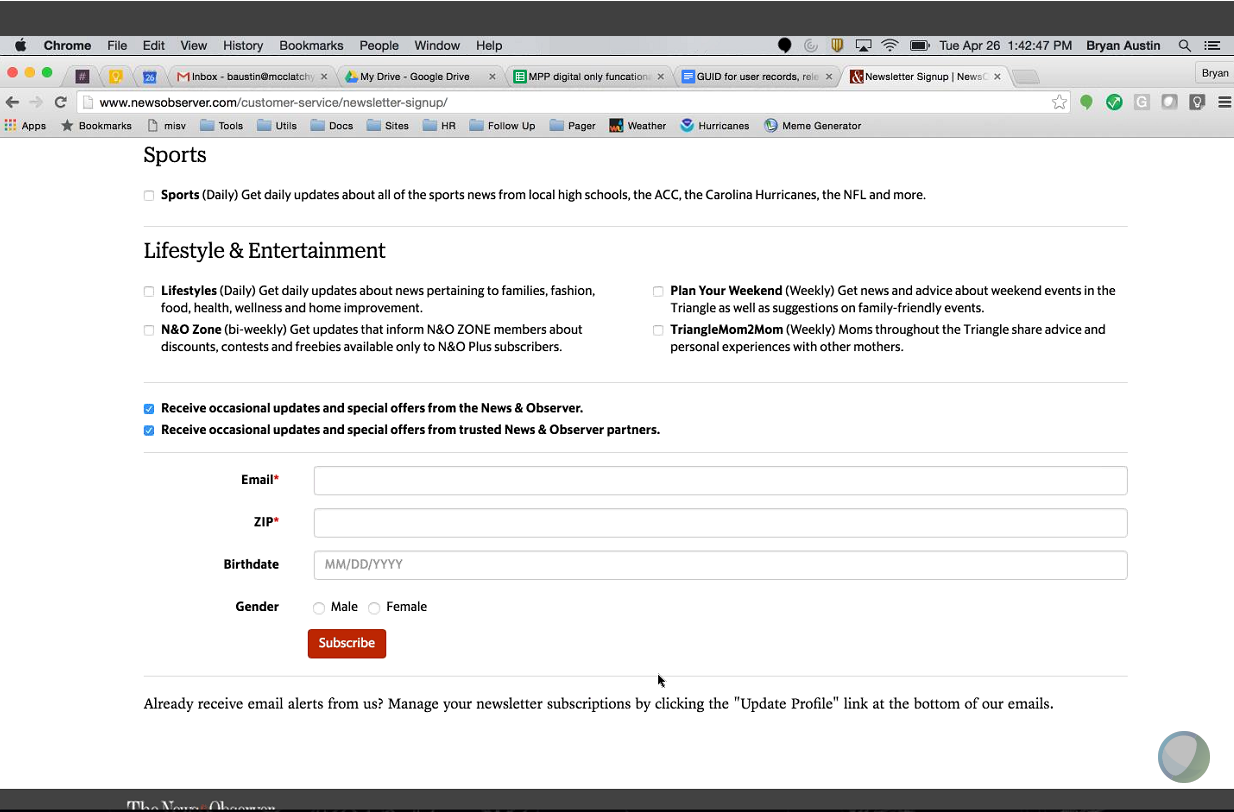
* Working MPP project
* Next V1 scope
* Dashboard – Loyalty information, account information,
* NCS and MPP integration – Technical constrains
* Link to I-service, NCS
* I-Service web site – API integration
* Two separates account NCS and MPP (subscription)
  + Information from NCS – basic information is pulled out (First Name, Last Name, email, address and Credit card to validate data integration)
* MPP – Stubbed out the Epic and Use cases – will be stubbing out the user stories
* Leanne to invite Wipro team for the user story planning meeting
* Apps are not of V1 release
* Push notification and SMS rollout – after Aug release
* Experian release before from MPP
* Dynamic pricing model as part of the release
* MPP provides plugin for Akamai and any clarification will be used via Jira (MPP system)
* Print subscription thought NCS only
* Market by Market transition (based on region) – product management group will take a decision on the rollout
* MPP is set-up on regional product wise
* Experian - Using email and new letters – several new letters are shared with customer (exact target) – breaking news
* Corn job enabled to send email, email via link sent to customer (breaking news)
* Email subscription are categorized like breaking news, sports etc.
* Automatic data movement to Experian – no there are some technical challenge which are been discussed with vendor
* Import data from different target and handover the file to system – working in progress, data structure discussion going on
* Anonymous users can get the new letter including subscribed users also.
* Experian enabled the API to interact with system to fetch data
* S-FTP (secured File transfer protocol) are used for file transfer
* Form post will be used
* Automation content or Manual content change will happened based on breaking news
* SMS –Global Mobile vendor for push
* Deal and coupon – deal saver
* Experian has report captured on email, or any other action information traced by a metrics
* Future roadmap will be MPP and Experian
* Enterprise market Parent - each market will be a child template

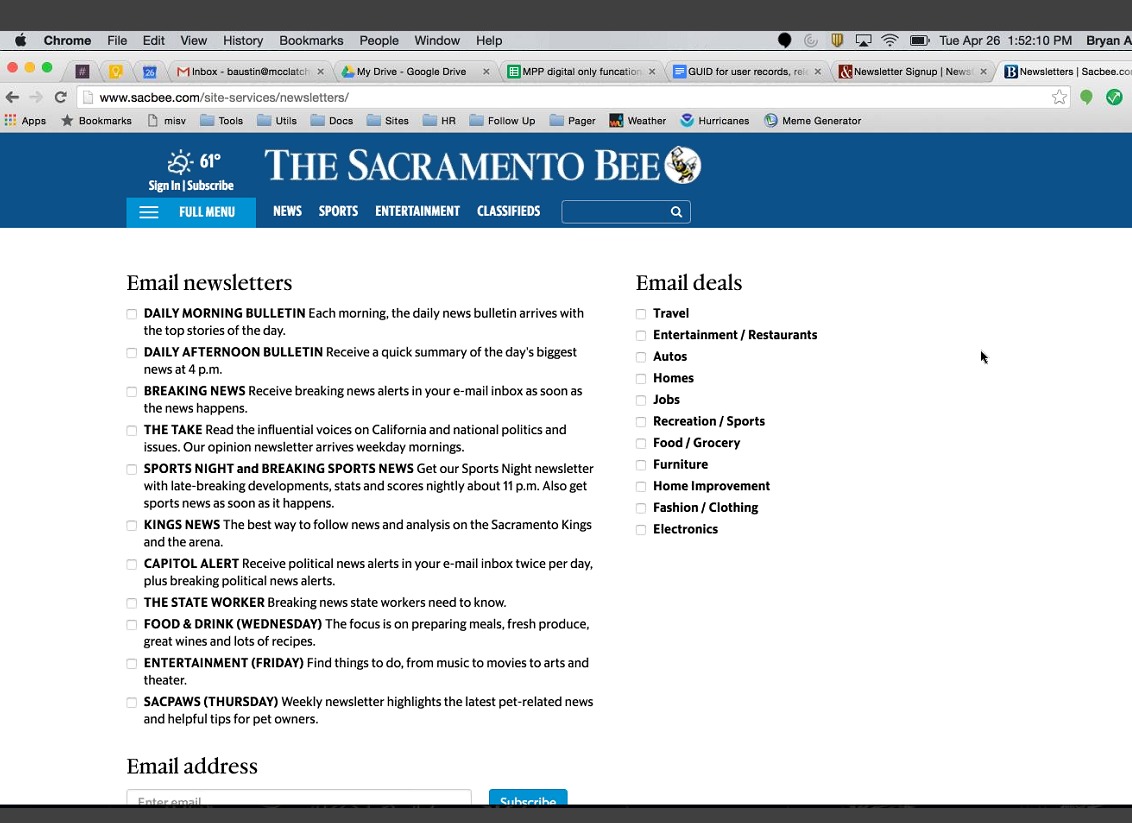










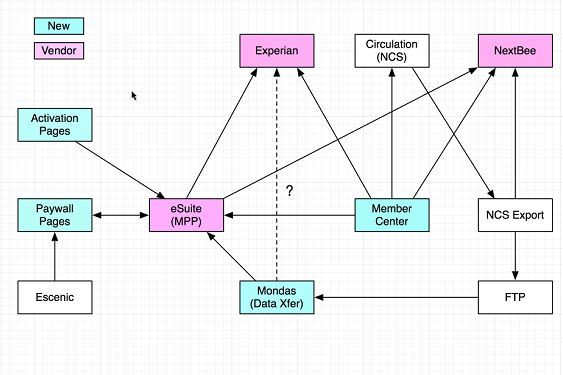


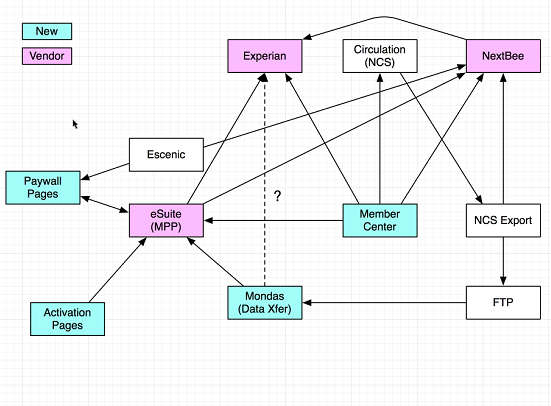


Olive:

* News gate to Olive – push gate
* Email goes without notification – E-Edition
* E-edition apps to stores
* Schedule job to send out notification
* Changes to Olive will interface with MPP







**Mobile**

Overall architecture (Core and Niche Apps, Hybrid Apps and Native Apps, Design Patterns) - do not have one, need to discover jointly

Joe to share the catalogs

Road map and features

Localization - Support English and Spanish

Personalization - New app do support

Videolicious is part of Escenic session, mobile team does not take care of this

API Integration - Joint session - Security, APIs used for Android and iOS, Caching, Architecture

Moving existing apps to universal apps - All moved

Aggregation API - still under discussion

Moving away from Hybrid App to Native App

Video Player - Use native player and move away from Youtube Player

Analytics - Omniture - Reports generated

Sprint to Deploy - step by step process

3rd Party tool - Volley used for networking, logger captured using tool

build.Gradle script

                Matrix build in Jenkins to build apps separately

                Not faced build error at least for an year

Walk through of build and deploy - Android - Dedicated session during KAP

3 Branches in GIT - Local build for developers, Integration testing (Develop) and Master for PROD. Rebases contains everything.

Need to have a dedicated session with Chris on Version management and Build process for both Android and iOS

Standard followed for comments during Checkin to JIRA

Single Code base repository - No Master/Slave

Build tool - PMD, Find Bugs and Check Styles - Need to have a dedicated session

Symbol files are stored in Crashlytics, last 3 APKs are stored.... No changes to the existing APKs and always a fresh build and deploy

Always single target - not multiple targets

Market targets - config.json - config to tell app from where to get the content, schema.xceed - xcode how to build the file and images - static images specific to apps

Deliver tool used for deploy apps to Apple

Need to have a dedicated session on Fastlane

Market specific external apps deployed using McClatchy account - deal saver, miami finder and miami flash

                Build by 3rd party and deployed by Chris. In case of issue(s), vendor will have to fix before deployment

**Mobile Testing:**

iOS no automation. Only automation they have for screen shots to app store. All manual testing.

Old apps - No testing strategy. We have to come up with the strategy document for a particular release.

Test cases - should be ready before the development actually starts

Regression happens on the tickets (Stories) for the specific release.

Every ticket (story) has testing estimated.

Tools -

                Tested on real devices - using AWS test bed (Appthwack) - Need a session on this on how the code is deployed and the testing is happening on Appthwack

                xccode ui testing for automation - need to have a dedicated session code walk through

                xceed test for ui

Regression -

                Old apps - it is all manual

                New App - need to build

                Green path test cases - no documentation to determine what test cases need to tested for each release. - Deep dive needed to come up with the documentation

                test data - testing team creates

SLA session is needed

Session on new release how upgrade happens

Lessons learned - dedicated session

Android

Better unit testing. Automation framework available - need a dedicated session.

New app - Expresso - UI test automation. - need to have a dedicated session on ui automation.

Tools - Unit testing Android junit.

Devices - Samsung and Nexus

Screen shot - Sebastin to check with the team whether this is happening

Configuration file is generated on the server side?

**Apps Go Live Plan**

New apps - Sep Oct time frame... not finalized....

Existing apps - only maintenance is happening, go live plan is not rigid and no specific day of the week for release (avoid Friday)

Severity level is in JIRA this is for Escenic, for Mobile decision taken based on discussion with different team (Joe, Walter few others)

Warranty period - no such concept here... based on the severity it is handled... hot fixes are for critical/major issues.

Overall depency plan - Across all towers from Release/Sprint point - dedicated session

Dedicated session on how release happens during certain non-planned news - configuration files are used to handle

**Testing**

Test data preparation & maintenance  - There is no separate team to enable test data for testing, individual QA team has to update test data while creating automation script. Also when cross functional testing they consume existing data to verify the scenarios. We have a separate session to discuss on Test Data Management..

1. Test environment ownership, deployment & co-ordination – There are three environments QA1, QA2 and Alpha, we have two session, 1. understand on Test environment and 2.Process on Build and Deployment
2. QA/Testing entry and Exit criteria – official there is no entry or exit criteria from testing, Going forward we need to formalize the process.
3. QA Gating sign-off process – Same as point 3

Note on Testing:-

* Testing team does not have any manual test cases for each story/Task – some of the requirement will have UAT scenarios in the JIRA against the story.
* Available UAT Scenarios will be considered for automation. There is no traceability or mapping to automation scripts to requirement
* Automation scripting has been enabled using Ruby/Cucumber tool – used for Uniting testing, UI Testing and API testing
* The Automation Framework is driven using Keyword and Modular Approach.
* Automation scripts does validation with both positive and negative scenario
* Currently Testing is done on 3 different browser and automation scripts are enabled for 3 browser but it can be expanded to additional browser.
* Bridge testing is done manual as there is no access to Newgate to enable automation, There are some plan in furture
* There is no specific testing on NFR, all are handled by dev team using JSP tool and Dynatrace tool
* Automation scripts are kicked off using Jenkin’s tool – Smoke test and Regression test suite for all QA environment and Production
* Jira is the defect management tool leverage for all modules
* Test Management is not part of Jira – (Creating Test plan, Execute test plan, Etc.)
* No testing metric are enabled
* No official sign off from testing team, once the Jira ticked is testing, development will take a decision to release to production
* Testing efforts are not tracked for all testing activities
* Smoke test and regression suite are updated based on the Jira ticket, they don’t follow any approach.