Philadelphia Eagles

The Evolution of Integrations Supporting a Championship Team and their Growth in Media
“Traditional” Integrations

Without a core platform for integration multiple systems become bottlenecks

Filesystems become “orchestrators”

Adoption of new systems is overly complex

One system failure affects multiple systems

Support is unpredictable and complicated. Your documentation must be flawless.

“Accidental architectures”
Services Driven Integrations

Core integration platform acts as the orchestrator of your workflows

Edge systems can change out as needed

Time to implement new integrations and workflows is lessened significantly

End user training streamlined by common interfaces and methods

Support streamlined through common languages, views, and reporting
Creating an Integration Plan
Creating an Integration Plan

• Themes from planning in 2014

  • They ask for a MAM but describe an Automation Platform

  • Every discussion ended in multiple “I want a pony” requests

  • The Executive Producer had many entrepreneurial ideas

  • Systems and users would be moving physically

  • IT required high security along with simplicity in support
Create an Integration Plan

• Our requirements for success
  
  • Quick implementation of new capabilities during short off-seasons
  
  • Compatibility with XML, JSON, REST, SOAP, Hotfolders, etc
  
  • Single pane of glass workflow status reporting
  
  • On Premises ESB capable of vertical and horizontal growth in integrations
  
  • Extension of workflow integrations into cloud systems
Infrastructure Overview
Infrastructure Overview

- Security requirements demanded core systems be on premises
- Primary set of users located at NovaCare Center
- Gameday users located at Lincoln Financial Field
- Required user experiences at both locations to support reorganizations and scaling on the fly
- Development Environment crucial to design, testing, and adoption of new integrations. Also serves as failover site with replicated DBs
Growth and Evolution of Integrations

Developing over 150 custom workflows over 4 years
Growth and Evolution of Integrations

• 2014
  • Ingest Historical content
  • Ingest Gameday content
  • Ingest Event/Community Content
  • Automate ingest replication to DR
  • Deliver to Broadcasters via FTP
  • Deliver to Comcast VOD via API
  • Export to Tricasters
  • Deliver to team website
  • Open asset on remote machine

• 2015
  • Add bugs to video exports
  • Adopt new panel integrations
  • Deliver to flyeaglesfly.com via API
  • Deliver to new broadcaster via API
  • Ingest Images from Photoshelter
  • Ingest Galleries from Photoshelter
  • Deliver to YouTube via API
  • Export Podcast audio

• 2015 (cont.)
  • Resize images
  • Reassign all metadata in new metadata schema

• 2016
  • Ingest Gameday content with gameday metadata
  • Check-in/out design images with managed renditions
  • Deliver for Twitter/Facebook/Vine/Instagram
  • Ingest from MediaShuttle
  • Deliver to YouTube via updated APIs
  • Rename files on ingest
  • Archive/Restore assets and collections to LTO
  • Manage file deletion by Managers
  • Track collection total size online and archived
  • Deliver to Photoshelter Galleries
  • Sync Photoshelter Galleries with Collections
  • Automate Ingest from Tricasters
  • Download whole collections

• 2017
  • Adopt new platform version, Uis, and Desktop app
  • Extend access remotely through secure cloud based UI and S3 file exchange
Choose a Platform that Supports Your Plan

Don’t end up with accidental architectures
Common Integration Language

• All workflows were written in Spring XML

• Serves as the instruction set to the platform

• Online XSD provides on-the-fly schema validation

• Leverage java functions easily from the platform

• Create case statements and branching

• Control workflow step progress, labels, and errors
Agility through Integration

- API calls defined as steps in the Workflow XML
- Store responses and results in variables to pass into future steps or subflows
- Call on keys stored securely outside of the workflow
Agility through Integration

- Extend workflows with Groovy

- Age old battle over scripting methods and languages is nearly eliminated

- Now when you go outside the norm it can be for an agreed reason, and supported
Agility through Integration

Automated branching allows for less decision making on the users

Error handling allows for ease of support
Agility through Integration

Workflow step queueing allows for management of system resources and endpoint bandwidth or connection requirements.
End User’s Experience

Streamlining UI/UX and Support to cater to non-technical users, seasonal interns, and high pressure environments
Streamlined Workflow Choices
Streamlined Workflow Choices

Combine multiple business processes into one user facing workflow for efficiency.

Add new delivery endpoint options that drive subflow branches behind the scenes, users only have to learn one primary workflow process while it continually expands.
Streamlined Support

Successful jobs in a single pane of glass validate all aspects of the workflow and business process for the user and administrators.
Streamlined Support

Control failures through clearly labeled steps, error handling, and user notifications.
Streamlined Support - Documentation

• “It takes a village”
  • IP Sheets
  • Shared password vaults
  • AD Integration across all products/services
  • Connectivity diagrams
  • Traffic and Services maps
  • Workflow swimlanes
  • Monitoring services and statistics
  • Storing statistics for frequent analysis
  • Partner/Integrator relationships, not just your vendors
Streamlined Support - Documentation

- Lay out traffic maps for infosec approval and network design
Streamlined Support - Documentation

- Monitor for anomalies and thresholds
- Automate alerts for systems with defined watermarks
- Gather statistics without alerts for systems without defined watermarks
- Choose a monitoring system that includes BPM rule sets
Streamlined Support - Documentation

• Create Swimlanes of business process AND of technical workflows

• Create lanes per service and/or per server, related to traffic and services maps

• Create swimlanes *before* integration, be sure to finalize *after* integration
What does success look like?

A *lot* of green workflow status indicators, and trophies...
Key Metrics

- Peak of over 10,000 workflows run during NFC Championship week, not including subflows
  - Approximately 1500 workflows during the game, *not including subflows*
  - Over 7500 new assets the week of the NFC Championship game
  - 90% of workflows are ingests, with 50% of ingests are automated (dreamcatcher, photoshelter, mediashuttle) including metadata and cataloging
  - 865 photos checked out, modified, and checked back in during the game before distribution to social media and apps
2018 and Beyond
Perpetual Modernization, and more Lombardi Trophies?
Integration continues…

• Replace Photoshelter workflows with Libris

• Onboard additional users (Executive, PR, Marketing), driving additional workflow needs

• Potentially move the application to the AWS – streamline home and away experiences
  • Add workflow gateway servers to on-prem locations to interact with gameday systems and production storage
  • Proxy streaming from S3 for consistent access from any location
  • Adding archive to Glacier adds another layer of data protection and less effort on staff IT
Thank You

CHESA
chesapeake systems