

# Content Scheduling System Documentation

## Executive Overview

The content scheduling system is a sophisticated automated broadcast scheduling solution that creates daily and weekly schedules by intelligently selecting and rotating content based on multiple factors including content metadata, scheduling rules, and broadcast requirements. The system ensures diverse, fresh content while respecting business rules around content expiration, replay delays, and thematic variety.

## Key Features

- **Automated Content Selection:** Intelligently selects content based on multiple scoring factors
- **Rotation Management:** Ensures variety through duration-based rotation and replay delays
- **Featured Content Support:** Prioritizes important content with special scheduling rules
- **Expiration Management:** Automatically imports and respects content expiration dates from Castus servers
- **Theme-Based Diversity:** Prevents repetitive themes in short-form content
- **Day/Night Scheduling:** Adapts content selection based on time of day
- **Weekly Template Support:** Creates consistent weekly schedules with imported meeting content
- **Enhanced Package Visibility:** Expanded rotation system to better highlight package content

## Technical Architecture

### Database Schema

The system uses PostgreSQL with several key tables:

1. **assets** - Master content records
  - **id:** Primary key
  - **content\_type:** Type code (AN, BMP, PSA, MTG, PKG, etc.)
  - **content\_title:** Display name
  - **duration\_seconds:** Content length
  - **duration\_category:** Calculated category (id, spots, short\_form, long\_form)
  - **engagement\_score:** AI-calculated viewer engagement metric
  - **theme:** Content theme for diversity tracking
2. **scheduling\_metadata** - Scheduling-specific data
  - **asset\_id:** Foreign key to assets
  - **last\_scheduled\_date:** When last aired
  - **total\_airings:** Cumulative play count
  - **featured:** Boolean for priority content
  - **content\_expiry\_date:** When content expires
  - **go\_live\_date:** When content becomes available
  - **available\_for\_scheduling:** Manual override flag
3. **scheduled\_items** - Individual schedule entries
  - Links assets to specific schedule times
  - Tracks actual air times for replay delay calculations

## Content Selection Logic

### 1. Enhanced Duration-Based Rotation System

The scheduler uses a configurable rotation order that has been expanded to better highlight package content:

```
# Original rotation
DEFAULT_ROTATION = ['id', 'spots', 'short_form', 'long_form']

# Enhanced rotation for better package visibility
ENHANCED_ROTATION = ['id', 'spots', 'pkg', 'short_form', 'pkg', 'long_form']
```

This enhancement ensures that package content (PKG) appears more frequently in the rotation cycle, giving it dedicated slots between other content categories. The impact:

- Packages get 2 guaranteed slots per rotation cycle
- Prevents packages from being buried in general short\_form category
- Maintains overall schedule balance while increasing package visibility

### 2. Content Filtering Pipeline

When selecting content, the system applies these filters in order:

#### a. Availability Filters

```
-- Must be analyzed and available
WHERE a.analysis_completed = TRUE
AND COALESCE(sm.available_for_scheduling, TRUE) = TRUE
```

#### b. Date Range Filters

```
-- Content must be within its valid date range
AND COALESCE(sm.content_expiry_date, '2099-12-31') > [schedule_date]
AND (sm.go_live_date IS NULL OR sm.go_live_date <= [schedule_date])
```

**c. Replay Delay Enforcement** The system enforces minimum time between replays:

- **Regular content:** Base delay + (total\_airings × additional\_delay)
- **Featured content:** Fixed minimum spacing (default 2 hours)

Example for spots category:

- Base delay: 48 hours
- Additional delay: 2 hours per airing
- After 5 airings:  $48 + (5 \times 2) = 58$  hours required

**d. Progressive Delay Reduction** When content is scarce, the system progressively reduces delays:

1. 100% of configured delay (normal)
2. 75% of configured delay
3. 50% of configured delay
4. 25% of configured delay
5. 0% delay (emergency mode)
6. Category reset (clears all delays for that duration category)

### 3. Content Scoring Algorithm

Content is scored using a weighted multi-factor system:

```

score = (
    freshness_score * 0.35 +      # How recently encoded
    engagement_score * 0.25 +     # AI-determined viewer interest
    scheduling_frequency * 0.20 + # How often it's been aired
    time_since_last_air * 0.20   # How long since last played
)

```

**Scoring Components:** **Freshness Score** (35% weight): - < 1 day old: 100 points - 1-3 days: 90 points - 3-7 days: 80 points - 7-14 days: 60 points - 14-30 days: 40 points - > 30 days: 20 points

**Engagement Score** (25% weight): - Direct pass-through of AI-calculated score (0-100)

**Scheduling Frequency** (20% weight): - Never aired: 100 points - 1-2 times: 80 points - 3-5 times: 60 points - 6-10 times: 40 points - 11-20 times: 20 points - > 20 times: 10 points

**Time Since Last Air** (20% weight): - Never aired or >24 hours: 100 points - 12-24 hours: 80 points - 6-12 hours: 60 points - 3-6 hours: 40 points - 1-3 hours: 20 points - < 1 hour: 0 points

#### 4. Impact of Parameters on Content Diversity

The system's ability to maintain content diversity is heavily influenced by several key parameters:

##### Available Content Pool Size

- **Large pool (100+ items per category):** Excellent diversity, natural rotation
- **Medium pool (50-100 items):** Good diversity, occasional repeats after several days
- **Small pool (20-50 items):** Moderate diversity, daily repeats likely
- **Minimal pool (<20 items):** Poor diversity, frequent repeats, delay reductions triggered

##### Replay Delay Settings

- **Longer delays:** Force more diversity but risk content exhaustion
- **Shorter delays:** Allow more repeats but ensure schedule completion
- **Progressive reduction:** Maintains quality as long as possible before compromising

##### Content Duration Distribution

- **Balanced distribution:** Smooth rotation through all categories
- **Skewed distribution:** May cause certain categories to repeat more frequently
- **Missing categories:** Forces system to skip rotation slots, reducing diversity

##### Feature Flags Impact

- **More featured content:** Reduces overall diversity due to priority scheduling
- **Featured content spacing:** Lower values (e.g., 2 hours) allow more repeats
- **Daytime bias:** Concentrates featured content, affecting time-based diversity

## 5. Theme Conflict Prevention

For short-duration content (IDs and spots), the system prevents consecutive items with the same theme:

```
if (last_category in ['id', 'spots'] and
    current_category in ['id', 'spots'] and
    current_theme == last_theme):
    # Skip this content and try next
```

This ensures variety in messaging for frequently-played short content.

## 6. Featured Content Handling

Featured content receives special treatment:

1. **Reduced Replay Delays:** Fixed 2-hour minimum instead of category-based delays
2. **Priority Selection:** Selected immediately when encountered
3. **Daytime Preference:** 75% chance of being scheduled during daytime hours (6 AM - 6 PM)
4. **Auto-Featuring:** Content can be automatically featured based on:
  - High engagement scores (>70)
  - Meeting content within 3 days of meeting date
  - Manual featured flag in database

## 7. Content Type Specific Rules

Different content types have unique scheduling parameters:

### Meeting Content (MTG)

- Auto-featured for 3 days after meeting date
- Higher replay delays (8-hour base)
- Maximum 4 plays/day when fresh, 2 when relevant, 1 when archived

### Packages (PKG)

- Dedicated rotation slots for increased visibility
- Moderate replay delays (3-hour base)
- Balanced between news value and variety needs

### PSAs and Announcements (PSA, AN)

- Shorter replay delays (2-hour base)
- Theme conflict prevention
- Higher rotation frequency

### Bumpers (BMP)

- Day/night specific scheduling
- Very short durations

- High replay frequency allowed

## Castus Integration

### Expiration Date Import

Every 3 hours, the system synchronizes with Castus servers:

1. **Connects to Castus API** using stored credentials
2. **Retrieves asset metadata** including expiration dates
3. **Updates scheduling\_metadata** table with latest expiry dates
4. **Logs changes** for audit trail

```
# Runs on schedule
sync_all_castus_expirations()
# Updates content_expiry_date for all matched assets
```

This ensures content automatically expires according to Castus-managed rights windows.

## Weekly Schedule Generation

### Process Flow

1. **Import Weekly Meetings**
  - User uploads schedule template (PDF/Excel)
  - System extracts meeting times and durations
  - Creates template with meeting blocks
2. **Fill Template Gaps**
  - Identifies empty time slots
  - Uses standard content selection logic
  - Respects meeting boundaries
3. **Apply Day/Night Rules**
  - Different content preferences for daytime vs nighttime
  - Bumpers scheduled appropriately
  - Featured content biased toward daytime

## Template Management

Templates can be: - **Saved**: Store recurring meeting patterns - **Loaded**: Apply saved patterns to new weeks - **Modified**: Adjust for special events - **Exported**: Share with other systems

## Advanced Features

### 1. Gap Filling Algorithm

When filling schedule gaps:

1. Calculate gap duration
2. Find content that fits without exceeding boundaries
3. Apply all standard selection criteria
4. Prefer content that closely matches gap size
5. Handle midnight boundaries specially

## **2. Emergency Content Handling**

When no content meets criteria: 1. Progressive delay reduction 2. Category-specific resets 3. Cross-category borrowing 4. Manual intervention alerts

## **3. Schedule Quality Metrics**

The system tracks: - Content diversity score - Theme repetition rate - Average content age - Replay frequency distribution - Gap fill efficiency

### **Reporting and Analytics**

#### **1. Content Diversity Report**

The Content Diversity Report provides comprehensive analysis of schedule variety:

**Key Metrics:** - **Unique Content Ratio:** Percentage of unique assets vs total slots - **Repeat Frequency Distribution:** How often each asset appears - **Category Balance:** Distribution across duration categories - **Theme Diversity Score:** Measure of thematic variety - **Time-based Diversity:** Variety analysis by daypart

**Report Sections:** 1. **Executive Summary** - Overall diversity score (0-100) - Key findings and recommendations - Comparison to previous periods

#### **2. Detailed Analysis**

- Content repeat patterns
- Most/least played content
- Category rotation effectiveness
- Theme clustering analysis

#### **3. Diversity Trends**

- Daily diversity scores
- Weekly patterns
- Impact of content pool changes

#### **4. Recommendations**

- Content gaps to fill
- Over-scheduled content to reduce
- Optimal replay delay adjustments

## **2. Schedule Analysis Report**

Provides detailed breakdown of schedule composition: - Content type distribution - Duration category percentages - Featured content placement - Meeting vs regular content ratio

## **3. Content Performance Report**

Tracks individual content metrics: - Play count history - Engagement score trends - Time since last air - Expiration warnings

## 4. Replay Pattern Analysis

Examines content repetition: - Replay delay compliance - Emergency mode frequency - Delay reduction events - Category reset occurrences

## 5. System Health Report

Monitors scheduling system performance: - Query execution times - Content pool sizes by category - Failed scheduling attempts - Manual intervention requirements

### Configuration

#### Key Settings (`scheduling_settings.json`)

```
{  
    "rotation_order": ["id", "spots", "pkg", "short_form", "pkg", "long_form"],  
    "replay_delays": {  
        "id": 24,  
        "spots": 48,  
        "short_form": 72,  
        "long_form": 72,  
        "pkg": 36  
    },  
    "featured_content": {  
        "daytime_hours": {"start": 6, "end": 18},  
        "daytime_probability": 0.75,  
        "minimum_spacing": 2  
    },  
    "meeting_relevance": {  
        "fresh_days": 3,  
        "relevant_days": 7,  
        "archive_days": 14  
    },  
    "diversity_targets": {  
        "minimum_unique_ratio": 0.3,  
        "maximum_repeat_count": 8,  
        "theme_variety_threshold": 0.6  
    }  
}
```

### Content Type Defaults

Each content type has specific scheduling parameters: - Replay delays - Maximum daily plays - Engagement thresholds - Auto-feature rules

## **Best Practices**

### **1. Content Preparation**

- Ensure accurate duration metadata
- Assign meaningful themes
- Set appropriate content types
- Configure expiration dates in Castus
- Maintain balanced content pools

### **2. Schedule Management**

- Review daily schedules for quality
- Monitor content expiration warnings
- Maintain adequate content inventory
- Use featured flags strategically
- Check diversity reports regularly

### **3. Performance Optimization**

- Regular database maintenance
- Index optimization on scheduling tables
- Periodic cleanup of old schedules
- Monitor query performance
- Balance content pool sizes

### **4. Diversity Optimization**

- Maintain 50+ items per category minimum
- Use appropriate replay delays for content value
- Monitor diversity scores weekly
- Adjust rotation order based on content mix
- Review and update themes regularly

## **Troubleshooting**

### **Common Issues**

#### **1. “No available content”**

- Check expiration dates
- Verify replay delays
- Review content inventory
- Check availability flags

#### **2. Poor Diversity Scores**

- Increase content pool size
- Adjust replay delays
- Review featured content settings
- Check rotation order configuration

#### **3. Theme Repetition**

- Ensure themes are properly tagged
- Check theme conflict logic
- Review short-form content variety

#### 4. Schedule Gaps

- Verify content duration distribution
- Check boundary handling
- Review gap-filling logs

### Debug Tools

- Detailed logging at each selection step
- Delay factor tracking
- Reset operation logging
- Content scoring breakdown
- Diversity metric calculations

### Future Enhancements

Potential improvements under consideration:

- Machine learning for optimal scheduling
- Viewer preference integration
- Dynamic replay delay adjustment
- Advanced theme clustering
- Real-time schedule adjustments
- Predictive content diversity modeling
- Automated content pool rebalancing