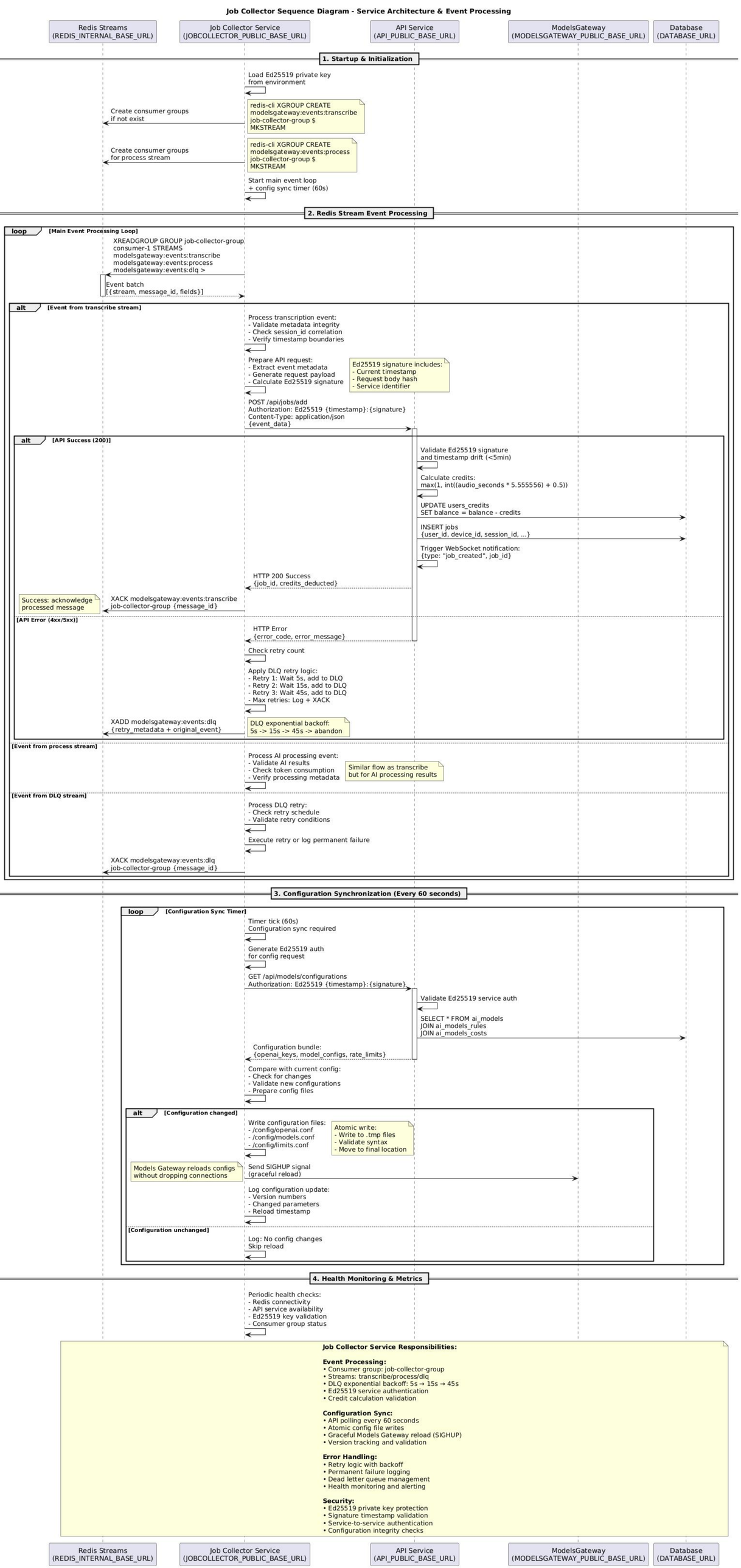
Job Collector Sequence Diagram - Service Architecture & Event Processing
Redis Streams (REDIS_INTERNAL_BASE_URL)
Job Collector Service (JOBCOLLECTOR_PUBLIC_BASE_URL)
API Service (API_PUBLIC_BASE_URL)
ModelsGateway (MODELSGATEWAY_PUBLIC_BASE_URL)
Database (DATABASE_URL)
1. Startup & Initialization
Load Ed25519 private key from environment
Create consumer groups if not exist
redis-cli XGROUP CREATE modelsgateway:events:transcribe job-collector-group $ MKSTREAM
Create consumer groups for process stream
redis-cli XGROUP CREATE modelsgateway:events:process job-collector-group $ MKSTREAM
Start main event loop + config sync timer (60s)
2. Redis Stream Event Processing
loop [Main Event Processing Loop]
XREADGROUP GROUP job-collector-group consumer-1 STREAMS modelsgateway:events:transcribe modelsgateway:events:process modelsgateway:events:dlq >
Event batch [{stream, message_id, fields}]
alt [Event from transcribe stream]
Process transcription event:
- Validate metadata integrity
- Check session_id correlation
- Verify timestamp boundaries
Prepare API request:
- Extract event metadata
- Generate request payload
- Calculate Ed25519 signature
Ed25519 signature includes:
- Current timestamp
- Request body hash
- Service identifier
POST /api/jobs/add
Authorization: Ed25519 {timestamp}:{signature}
Content-Type: application/json
{event_data}
alt [API Success (200)]
Validate Ed25519 signature and timestamp drift (<5min)
Calculate credits:
max(1, int((audio_seconds * 5.555556) + 0.5))
UPDATE users_credits
SET balance = balance - credits
INSERT jobs
{user_id, device_id, session_id, ...}
Trigger WebSocket notification:
{type: "job_created", job_id}
HTTP 200 Success
{job_id, credits_deducted}
Success: acknowledge processed message
XACK modelsgateway:events:transcribe job-collector-group {message_id}
[API Error (4xx/5xx)]
HTTP Error
{error_code, error_message}
Check retry count
Apply DLQ retry logic:
- Retry 1: Wait 5s, add to DLQ
- Retry 2: Wait 15s, add to DLQ
- Retry 3: Wait 45s, add to DLQ
- Max retries: Log + XACK
XADD modelsgateway:events:dlq
{retry_metadata + original_event}
DLQ exponential backoff:
5s -> 15s -> 45s -> abandon
[Event from process stream]
Process AI processing event:
- Validate AI results
- Check token consumption
- Verify processing metadata
Similar flow as transcribe but for AI processing results
[Event from DLQ stream]
Process DLQ retry:
- Check retry schedule
- Validate retry conditions
Execute retry or log permanent failure
XACK modelsgateway:events:dlq job-collector-group {message_id}
3. Configuration Synchronization (Every 60 seconds)
loop [Configuration Sync Timer]
Timer tick (60s)
Configuration sync required
Generate Ed25519 auth for config request
GET /api/models/configurations
Authorization: Ed25519 {timestamp}:{signature}
Validate Ed25519 service auth
SELECT * FROM ai_models
JOIN ai_models_rules
JOIN ai_models_costs
Configuration bundle:
{openai_keys, model_configs, rate_limits}
Compare with current config:
- Check for changes
- Validate new configurations
- Prepare config files
alt [Configuration changed]
Write configuration files:
- /config/openai.conf
- /config/models.conf
- /config/limits.conf
Atomic write:
- Write to .tmp files
- Validate syntax
- Move to final location
Models Gateway reloads configs without dropping connections
Send SIGHUP signal (graceful reload)
Log configuration update:
- Version numbers
- Changed parameters
- Reload timestamp
[Configuration unchanged]
Log: No config changes
Skip reload
4. Health Monitoring & Metrics
Periodic health checks:
- Redis connectivity
- API service availability
- Ed25519 key validation
- Consumer group status
Job Collector Service Responsibilities:
Event Processing:
• Consumer group: job-collector-group
• Streams: transcribe/process/dlq
• DLQ exponential backoff: 5s → 15s → 45s
• Ed25519 service authentication
• Credit calculation validation
Configuration Sync:
• API polling every 60 seconds
• Atomic config file writes
• Graceful Models Gateway reload (SIGHUP)
• Version tracking and validation
Error Handling:
• Retry logic with backoff
• Permanent failure logging
• Dead letter queue management
• Health monitoring and alerting
Security:
• Ed25519 private key protection
• Signature timestamp validation
• Service-to-service authentication
• Configuration integrity checks
Redis Streams (REDIS_INTERNAL_BASE_URL)
Job Collector Service (JOBCOLLECTOR_PUBLIC_BASE_URL)
API Service (API_PUBLIC_BASE_URL)
ModelsGateway (MODELSGATEWAY_PUBLIC_BASE_URL)
Database (DATABASE_URL)