

令和3年度 秋期 応用情報技術者試験「スマート生け簀システム」音声解説用スクリプト

1. 導入：スマート養殖システムが目指す戦略的価値

皆さん、こんにちは。今回は令和3年度 秋期 応用情報技術者試験の午後問7で取り上げられた「スマート生け簀システム」を徹底解説していきます。近年、第1次産業へのIT導入が加速していますが、この養殖業のスマート化は単なる労働力の自動化に留まりません。真の戦略的価値は、AIによる**「データ駆動型の成育管理」にあります。具体的には、AIが「**給餌重量**」と「**給餌日時**」を決定することで、**無駄な餌を減らす**「給餌コストの最適化」と、**狙い通りの**「**成長スピードのコントロール**」**を同時に実現する点にあります。この高度な制御を支えるシステムがどのようなデバイスで構成され、データがどう流れるのか、まずは全体像を鳥瞰していきましょう。

2. システムアーキテクチャの鳥瞰：ハードウェアとデータの流れ

スマート生け簀は、現場のデバイスとサーバ側のAIが密接に連携するエッジコンピューティングの典型的なモデルです。主要な構成要素の役割を整理してみましょう。

- **給餌部（ローカルコントローラ）**：サーバからの指示を受け、給餌予定回数や終了判断のしきい値を自律計算して実行する、現場の司令塔です。
- **水中ステレオカメラ**：養殖魚を撮影し、サーバ側で「大きさ・形状・推定個体数」という成長指標を抽出するための基礎データ（動画）を提供します。
- **残餌検知部（集餌コーン・残餌センサ）**：生け簀の底部で、食べ残された餌（残餌）を1個ずつカウントし、給餌終了の判断材料を作ります。
- **水質センサ**：水温や酸素濃度、塩分濃度を計測し、成育環境の基盤データを収集します。これらは独立しているのではなく、「給餌の最適化」という一つの目的のために、現場（エッジ）で発生した事象をサーバへ送り、AIによる高度な判断を現場にフィードバックするというサイクルを形成しているのです。それでは、データの質に関わる設問1から深掘りしていきましょう。

3. 設問1解説：物理的制約とデータの完全性

設問1では、システムの論理設計と、現場の「物理的制約」をどう橋渡しするかが問われました。

(1) 付加すべきデータ：因果関係を特定する「日時」

サーバでAI分析を行う際、水質や成長データに必ず付加すべきなのは**「日時データ」です。なぜなら、「いつ、どれだけの餌を与えた結果、どのように成長したか」という因果関係**を特定するには、全てのデータが時系列で紐付いている必要があるからです。日時の情報がなければ、成長過程の変化や時間帯ごとの食べ具合の傾向を分析する戦略的価値が失われてしまいます。

(2) 単位時間（5分）待つ理由：沈下という物理現象

システム設計には物理空間の理解が欠かせません。図1を見ると、給餌部は海面にあります。残餌センサは生け簀の底にあります。投入された餌が魚に食べられず、海中を沈下して底部のセンサを通過するまでには、物理的なタイムラグが発生します。給餌直後に読み込むのではなく、**「残餌が残餌センサに到達するまでに時間を要するから」**こそ、5分という待機時間が必要になるのです。データの「質」の次は、扱うデータの「量」とリソースの課題を見ていきましょう。

4. 設問2解説：リソース制約下でのデータ処理戦略

現場のデバイスは、ストレージ容量という厳しい制約の中で大容量データを扱わなければなりません。

(1) 動画データサイズの計算

ステレオカメラで2分間撮影した際のデータ量を、以下のステップで算出します。

1. **1フレーム分の容量：** 画素数 $800 \times 600 \times 1$ 画素 24ビット(3バイト) = 1,440,000バイト
2. **左右2台分の1フレーム：** $1,440,000 \times 2 = 2,880,000$ バイト
3. **全撮影データの総計：** $2,880,000 \times 20$ フレーム/秒 $\times 120$ 秒(2分) = 6,912,000,000バイト
4. **単位換算：** ソースの注釈に基づき 10^9 バイト = 1GB とすると、6.912 GBとなります。小数第1位を四捨五入した解答は、**7GB** です。

(2) 必要な処理：ストレージとのギャップを埋める

給餌部のフラッシュメモリ容量はわずか2GBしかありません。7GBの動画をそのまま保存することは不可能なため、水中ステレオカメラ内で**「画像圧縮処理」**（または「動画データの圧縮」）を行い、データサイズを削減する必要があります。エッジ側での適切なリソース管理が不可欠であることを示す好例です。

5. 設問3解説：給餌終了判断と消費重量の算定ロジック

最後は、現場での判断基準となるアルゴリズムの論理構成です。

図3：給餌処理フローの穴埋め

- **a：カウント値** 5分間隔で「カウント値」を読み込み、Ct値（単位時間当たりのカウント値）を求めます。
- **b：しきい値以上** Ct値が増えるということは、「魚が餌を食べず、見逃された餌が底にたくさん落ちてきている」という事態を指します。つまり、Ct値が**「しきい値以上」**になったとき、魚が満腹で「食欲」がなくなったとデジタルに判断し、給餌を終了させるのです。

食べた餌の重量の算定式

算出ロジックは、「投入した総重量」から「残った総重量」を引くというものです。食べた餌の重量 = (c × d) − (e × f)

- a カウント値
- b しきい値以上
- c ク (給餌を行った回数)
- f ウ (1個の餌の重量) ※dには「イ (1回当たりの餌の重量)」、eには「オ (給餌終了時のカウント値)」が入ります。ここでポイントなのは、cに「給餌予定回数 (キ)」ではなく**「給餌を行った回数 (ク)」**を使う点です。Ct値の判断によって途中で給餌を打ち切る可能性があるため、実際に投入した回数をいなければ正確な重量は出せません。実機の動作を正確に反映したロジックが求められています。

6. 総括：システムアーキテクトに求められる視点

本問題を通じて、IoTシステム設計における3つの柱が見えてきました。

1. **物理現象の理解：** 餌が沈む時間などの現場の物理法則を、待機時間としてロジックに組み込むこと。
2. **リソース管理：** 大容量データはエッジで圧縮加工し、分析に必要なメタデータはクラウドへ送るという役割分担 (エッジコンピューティング) を行うこと。
3. **論理的アルゴリズム：** センサの数値から「食欲」という抽象的な状態を判定し、正確な数値を導き出すこと。これらの視点は、皆さんが実務でAI連携システムを設計する際にも、そのまま通用する普遍的なスキルです。応用情報技術者試験の午後問題は、現場の課題を擬似体験させてくれる貴重な教材です。今回学んだ「物理と論理の橋渡し」の感覚を、ぜひ今後の学習や実務に活かしてください。皆さんの努力が、素晴らしいエンジニアキャリアに繋がることを心より応援しています。お疲れ様でした！