

# KNXを軸としたオープン プロトコル時代のオートメ ーション設計思想

快適な住空間を実現するための  
統合設計アプローチ





# 本日お伝えしたいこと

## 設備の分断

住宅現場で起きている設備やアプリの分断、設計意図と運用のズレ

## 10年先の更新

長期運用を見据えた現実的な対応策

## 設計思想

オーブンプロトコルをどう設計思想として扱うか

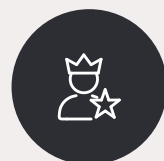
KNXの基礎解説や方式比較ではなく、実践的な設計思想を共有します。

# コンフォース株式会社紹介



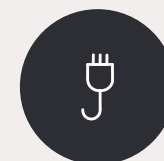
## 住宅オートメーション設計・施工

照明・空調・カーテン・AV・ネットワークなど、複数設備を横断した統合設計を提供します。



## 高級住宅・リノベーション対応

住まい全体の快適性と操作性を整える提案型オートメーションを実現します。



## マルチプロトコル対応

Lutron HomeWorks、KNX、国内空調（GW連携）、ネットワーク設計に対応しています。



# Comforce（コンフォース）

Comfortable Space（快適な空間）から発想した造語

住まいの快適さを、設備単体ではなく**空間全体として成立**させる  
思いを込めています



# KNXの魅力は"思想"にある



## 長く使える

長期運用に耐える安定性と持続性



## 運用が安定しやすい

分散アーキテクチャによる堅牢性



## メーカーに縛られにくい

特定メーカーの都合に振り回されない自由度



## 配線設計を積み上げられる

構造設計を丁寧に組み立てる基盤

**未来の変化を許容する構造** — これが住宅の長期運用と相性が良い理由です

# 適材適所の設計思想



## 照明

専門システムが得意な領域



## 空調

国内メーカー、ゲートウェイ経由が現実的



## センサー・入力整理

KNXが非常に得意な領域

- ❏ **重要:** KNXで全てをやる必要はありません。異なる要素同士の**橋渡し役**として、設計の軸に置くアプローチが実務では最も効果的です

# 住宅の「分断」という課題



『いまの住宅は、あまりにも分断されている』

- 照明は照明のアプリ
- 空調は空調のアプリ
- 音響は音響のアプリ
- IoTはIoTのアプリ

それぞれの操作体系が異なり、クラウド依存で、動作が遅れたり、家族の誰かが操作に戸惑うという現象が起きています。

# クラウド依存の問題



## クラウドが遅い

通知が来ない、反応が遅延する



## サービス終了リスク

クラウドサービスが終了する可能性



## 仕様変更

メーカーが仕様変更する不安定性

住宅の設備がクラウドに依存しすぎると、住む人にとっての「安定性」は失われてしまいます。さらに日本の場合、国内メーカーが独自方式を持っているという特徴があります。

住宅のスマート化は、**プロトコルの戦いではなく統合設計がテーマ**になるのです。



# 分断の三つの症状



## 反応速度と体感のズレ

押してすぐ反応するものと、1～2秒遅れて動くものが混在すると、「壊れているのでは」と感じる



## 例外条件の増殖

「このときは動くが、このときは動かない」が増えるほど、住む人は覚えきれなくなる



## 責任範囲の曖昧さ

何か不具合が出た際に「どこに連絡すればよいか」が分からないと、その機能自体が使われなくなる

# 分断の本質

分断の本質は、機器が増えること自体ではなく、暮らしの中で**"不安が増える構造"**になってしまいうことです。



## 設計意図の分断



## 施工の分断

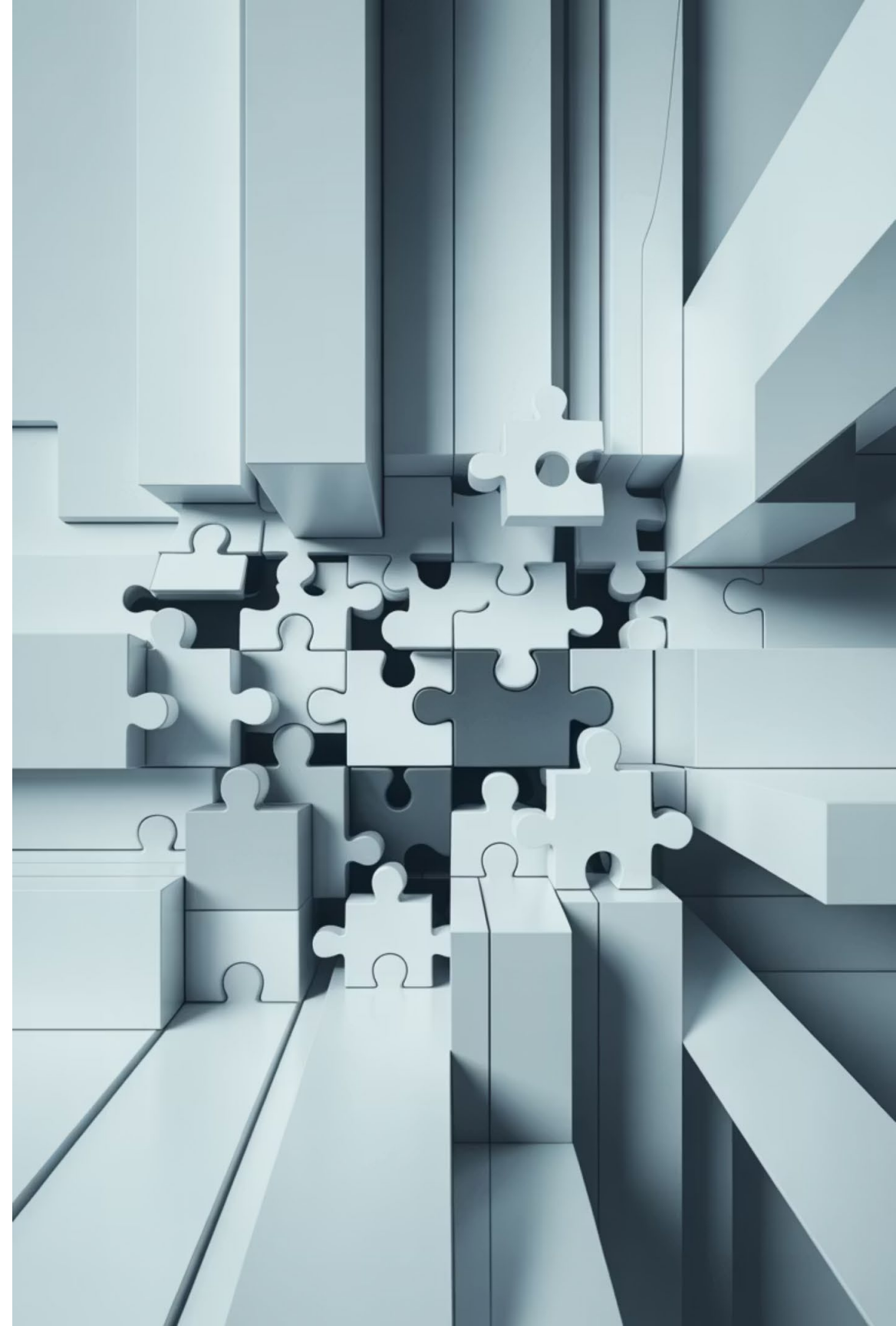


## 引き渡し後の分断



## 保守の分断

便利な部品が増えた結果、暮らしとしての**統一感が崩れる**。この崩れを整えるのが、インテグレーションであり、設計思想です。



# 住宅オートメーションの特徴



## 運用者がいない

ビルには管理者がいますが、住宅にはいません。操作は住む人全員。お子さんや高齢者も触ります。



## 意匠と生活動線の影響

見た目、操作位置、家具の配置など、生活と直結する要素が多いのが住宅です。



## リフォームで設備が混ざる


ビルの設備は一気に更新できますが、住宅は部分的な更新が多い。10年前の設備と最新機器が共存することも珍しくありません。



## 長期運用が前提

ビルは15～20年で大規模更新しますが、住宅は30年以上そのまま使われることもあります。

- ❑ だからこそ住宅は、プロトコルで一気に統一しようとしてもうまくいかないという特徴があります。そこで必要なのは、**複数プロトコルの共存を前提とした設計**です。



# 3層アーキテクチャ

## UIレイヤー

壁スイッチ、タブレット、音声など — ここで統一感が必要

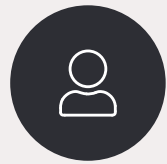
## 制御レイヤー

KNX、HomeWorks、空調GW — 各プロトコルが役割を持つ

## デバイスレイヤー

照明器具、エアコン、カーテン、AV機器などの実際の設備

# 設計作業：三つの表で整理



## UIの表

「誰が」「どこで」「何を」「  
どう操作するか」

玄関、LDK、寝室、夜間導線、  
来客時——状況で設計



## 制御（ルール）の表

トリガー、動作順序、例外条件  
を言語化

後から要望が増えても破綻しに  
くい



## デバイス（実体）の表

残る機器と更新されやすい機器  
配線、盤、接点、余裕を設計

---

統合は「全部を一つ」ではなく、境界線を決めて変化の影響範囲を小さくすること



# 統合ポイントの決定

この家の"勝ち筋"はどこか



## UI設計を厚くする

家族が簡単に使えることが  
最優先の場合



## 配線・構造を厚くする

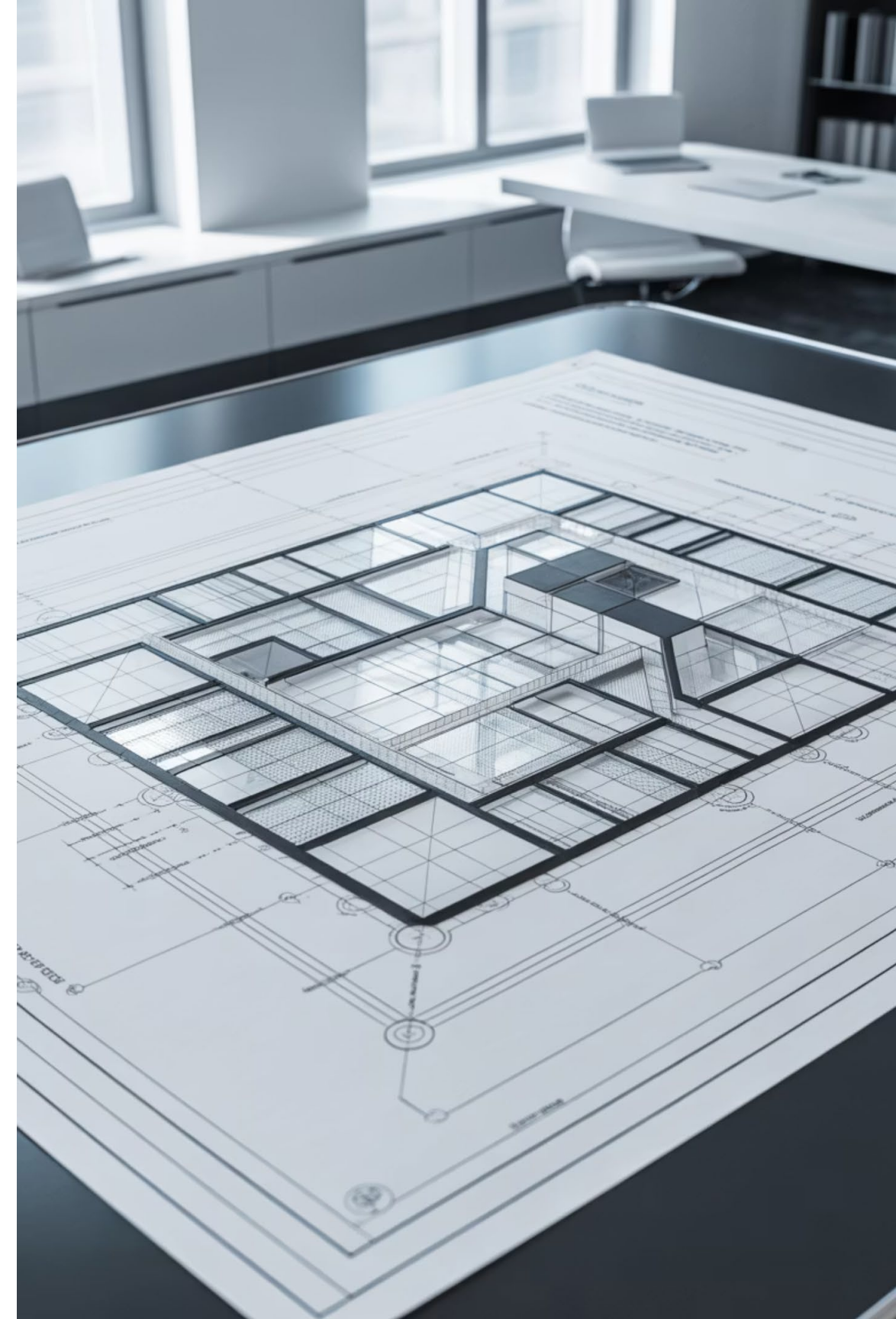
長期運用と更新耐性が  
最優先の場合



## 制御の整合性で勝つ

意匠指定が強い場合

**合言葉:** 統合とは、全部を一つにすることではなく、"境界線をはっきりさせること"です



# この家の「勝ち筋」を決める

## 家族の使いやすさ最優先

UI設計を厚くする

## 長期運用と更新耐性最優先

配線・構造・制御骨格を厚くする

## 意匠指定が強い

指定を尊重し制御の整合性で勝つ

- 統合とは、全部を一つにすることではなく、境界線をはっきりさせること"





# オープンプロトコル設計の 思想

オープンプロトコルの価値は、  
『他を否定しない自由』にあります。

KNXも、BACnetも、Modbusも、それぞれ良いところがあります。重要なのは、**これらを組み合わせて、目的に最適な設計にすること**です。

# 照明の扱い

## 指名／分業／設計プロセス

### 高級住宅での要件

- 壁スイッチの質感
- 空間に合わせたシーンの自由度
- 明るさや色味の変化の自然さ
- 操作体験まで含めた価値

### 実務での指定ケース

施主様・設計事務所・照明デザイナーから「このスイッチを使いたい」「この照明制御を前提に」といった指名が入ることも



□ **分業の明確化:** 光のデザインは専門家、制御の統合と動作の保証は私たち



# 照明で事故が起きやすいポイント

## 1 シーンの粒度

「どの器具が」「どの程度」「どのスピードで」変化するか——名前だけではなく、詳細を明確にする

## 2 例外条件

夜中にトイレへ行くとき、寝室は暗いままでよいのか。来客時は自動を止めたいのか。住宅は例外が多い

## 3 手動への戻し方

自動が効かないときに、誰でも迷わず戻せる導線を残す。これが結果的に信頼性につながる





# 空調の扱い

国内機器＋ゲートウェイ＋更新耐性

1

## 国内メーカー空調

固有仕様・制約が強い

2

## ゲートウェイ

プロトコルの翻訳機能

3

## 統合システム

扱いやすい形に整理

CoolMasterのような空調連携ゲートウェイで、統合システム側から扱いやすい形に整える

10年、15年のスパンで見れば買い替えが発生。**更新される部分**  
**だけ切り出して入れ替えられる構造**が重要です

# 引き渡し後に"本当の価値"が出る

## 保守・ドキュメント・合意形成

1

### 統合ポイントの説明

UI／制御／デバイスのどこで何を統合しているか

2

### 回路・系統・アドレス

将来誰が見ても追える形式で

3

### 連動条件

何がトリガーで、何が動くか

4

### 例外条件

来客時、夜間、手動優先など

5

### "戻し方"

自動が不調でも暮らしが止まらない

設計と施工の中に"将来の自分への手紙"を残すつもりでドキュメント化します

# 暮らしの変化に対応する 引き渡し後に起きること

- お子さまの成長でリズムが変わる
- 在宅勤務が増える
- 夜間導線を調整したい
- 来客時だけ自動を止めたい

これらは故障ではなく、暮らしが前に進んだ  
結果です



# ネットワークとセキュリティ

## 住宅で破綻しない最低ライン

- **制御系の通信は常に安定**  
リアルタイム性と信頼性の確保
- **ネットワークは増設前提で余裕を**  
将来の拡張に備えた設計
- **外部アクセスはやり過ぎない**  
運用者がいないため、複雑化を避ける

オートメーションは「制御」だけではなく、「土台」も含めて設計です





# まとめ

## 住宅は分断が起きやすい

アプリ、クラウド、操作者の多  
様性

## 長期運用で設備更新が前提

10年、20年先を見据えた設計

## 階層でどこを統合するか 決める

UI／制御／デバイスの境界線を  
明確に

## 施主・設計の指定を尊重

分業で成立させる柔軟性

## オーブンプロトコルは武器

更新に強い構造を作るための思想





ご清聴ありがとうございました