



# Integration Practicalities

---

Nick Meredith  
Chief Architect - Altus

# Background

---



**Chief Architect at Altus**



**15+ years of experience building and delivering componentised solutions**



**Strongly believe in a componentised approach**







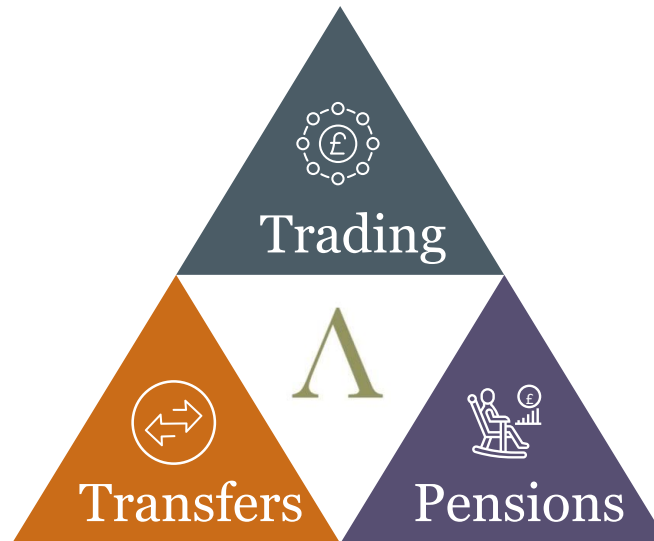
**Backed up by our own solutions**



# Altus Business Systems

---

-  **AIG / AFG - Trading Components**
-  **APG – Pension Payroll and HMRC Reporting Component**
-  **ATG – Account Transfers & Asset Re-Registration component**
-  **But... It does have its challenges**



# The Component Challenge

---



**It's a good architectural approach  
But... It does have its pitfalls**



**Often see poorly designed solutions**



**Regularly see the same mistakes**



**So what challenges do you face?**



# Technology Mismatch



## Modern vs Legacy



## Language / Platform differences

Python vs C# vs COBOL  
UNIX vs Windows



## Data encoding

XML vs JSON







## Differing Architectures

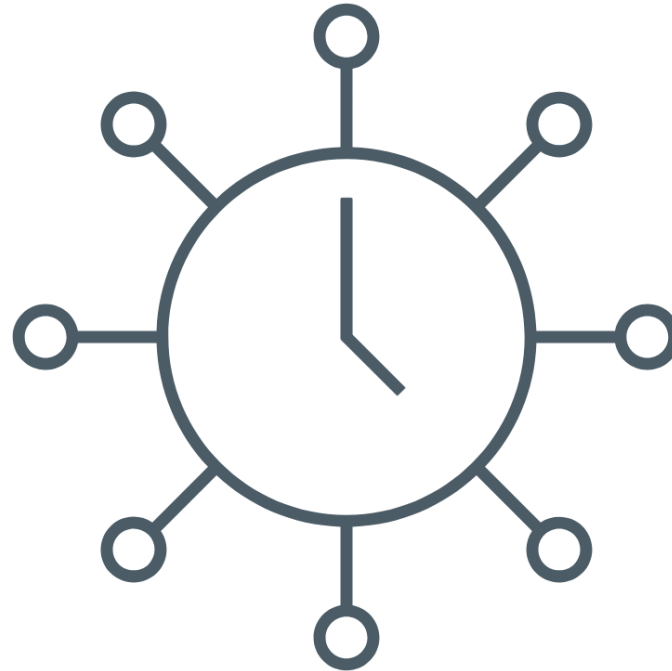
Client / Server vs web-based  
Service Bus vs P2P  
Hosted vs On Premise  
Pub/Sub vs Webservice



# Scheduling






---

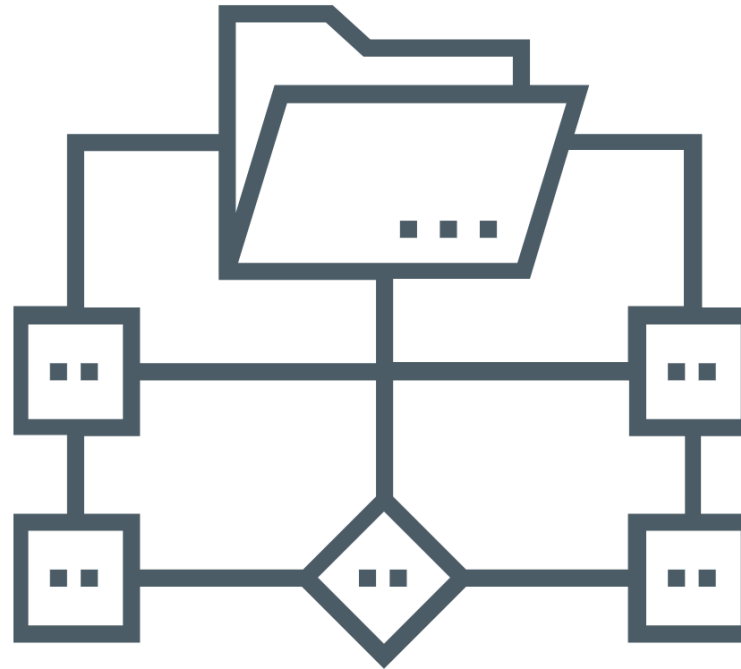
-  **Release Schedules**
-  **Processing Schedules**
-  **Downtime**
-  **Global businesses**



# Data





---

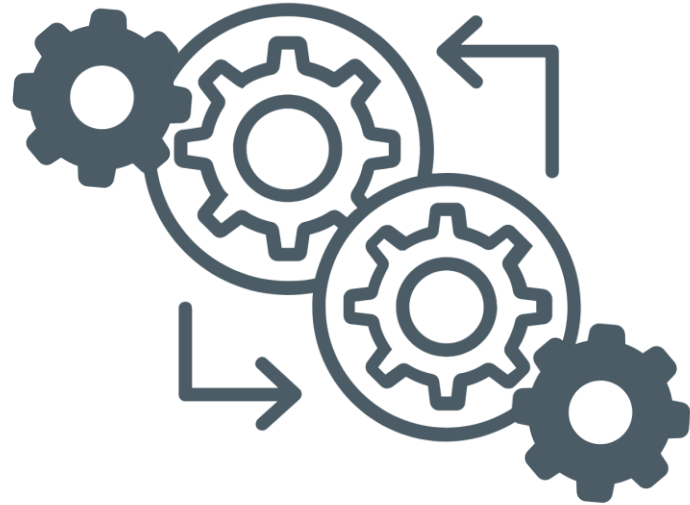
-  **Data Mapping**
-  **Data Interpretation**
-  **Master Data Management**
-  **Availability**
-  **Encoding**



# Process and Capability

---






-  **Batch vs Realtime**
-  **Functional Capability**
-  **Parallelism**
-  **Capability **Overlap****





# Management






---

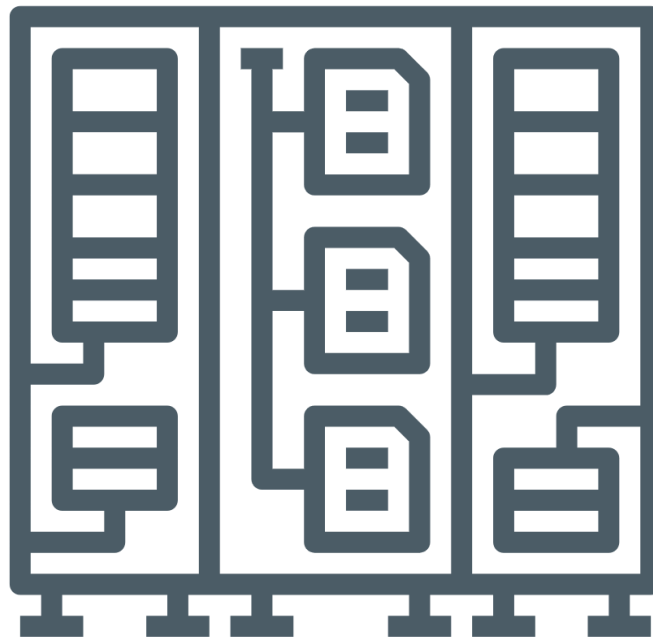
-  **Administration**
-  **DR / HA / Synchronisation**
-  **Support and Maintenance**
-  **MI Tracking / Debugging / Error Handling**
-  **Vendor Management**



# Usability







---

-  **Application Overload**
-  **Security**
-  **Accessibility**
-  **Source of Truth**
-  **Technology Driven solution**



# What's the Solution?

---

-  **It's hard... to get it right!**
-  **Data modelling / mapping**
-  **Process modelling / mapping**
-  **Orchestration**
-  **Obfuscation / Middleware**
-  **Lots of upfront thinking and it'll all be fine!**





Thank you

---