HEXAWARE

Autosys to Airflow

Introduction

Top of mind for customers using Autosys

- **78% of IT leaders** report their legacy workload schedulers like Autosys struggle to meet dynamic cloud scaling demands

- **70% of workload automation users** cite "lack of dynamic job scheduling" as a key limitation disrupting workflow agility.

- Customers find licensing and maintenance fees expensive, with **15-20% increase in YOY licensing cost**

- Requires specialized skills with **60% of organizations** report Autosys workflow scaling requires manual configuration

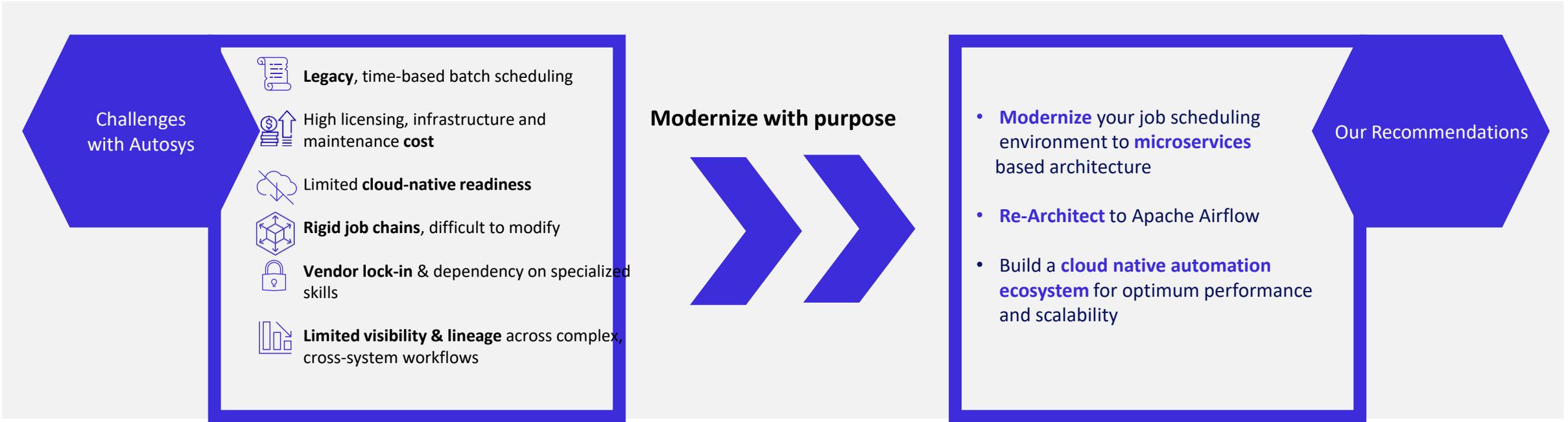## Autosys market insights in 2026

ISG

Provider Lens 2026

- **Limited dynamic job creation** and monitoring capabilities.

- Broadcom's multiple overlapping automation products **dilute enhancements for Autosys**

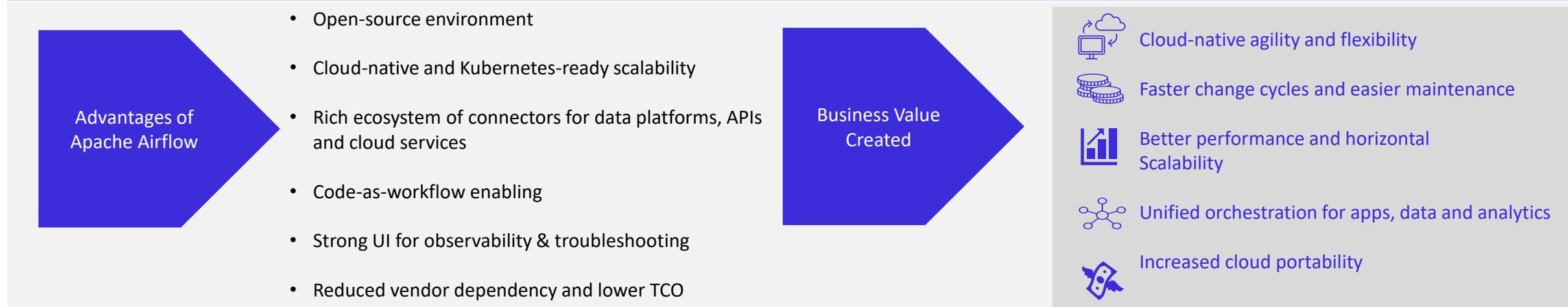- **Rising licensing costs** push users toward alternative solutions.

FORRESTER®

Predictions 2026 Report

- Legacy platforms like Autosys face challenges due to evolving enterprise priorities for **cloud native automation**

Future proof your business by moving from **Autosys to Airflow**

Challenges with Autosys

- **Legacy**, time-based batch scheduling
- High licensing, infrastructure and maintenance **cost**
- Limited **cloud-native readiness**
- **Rigid job chains**, difficult to modify
- **Vendor lock-in** & dependency on specialized skills
- **Limited visibility & lineage** across complex, cross-system workflows

Modernize with purpose

Our Recommendations

- **Modernize** your job scheduling environment to **microservices** based architecture
- **Re-Architect** to Apache Airflow
- Build a **cloud native automation ecosystem** for optimum performance and scalability

---

Why Moving from Autosys to Airflow is a Smart Business Strategy?

Advantages of Apache Airflow

- Open-source environment
- Cloud-native and Kubernetes-ready scalability
- Rich ecosystem of connectors for data platforms, APIs and cloud services
- Code-as-workflow enabling
- Strong UI for observability & troubleshooting
- Reduced vendor dependency and lower TCO

Business Value Created

- Cloud-native agility and flexibility
- Faster change cycles and easier maintenance
- Better performance and horizontal Scalability
- Unified orchestration for apps, data and analytics
- Increased cloud portability

Solution Summary

Our Goals

By modernizing legacy AutoSys-based workload to a cloud-native Apache Airflow platform, clients will **reduce the total cost of ownership by 30-50%** ,whilst reducing licensing and infrastructure costs, along with the advantage of enabling **scalable, resilient, and event-driven orchestration of enterprise workloads**. This transition also helps standardize and unify scheduling across application, data, and analytics pipelines while improving end-to-end visibility, governance, and operational control

Values Created

| Lower total cost of ownership by moving from proprietary AutoSys to open Airflow | Faster change and release cycles through code-driven, CI/CD-enabled workflows | Elastic scalability on Kubernetes or managed cloud Airflow for growing workloads | Higher reliability with automated retries, SLAs and intelligent failure handling | Unified monitoring, lineage and auditability across all enterprise workflows |

Key Solution Highlights

AMAZE-led ~70% automated conversion of AutoSys jobs into Airflow DAGs
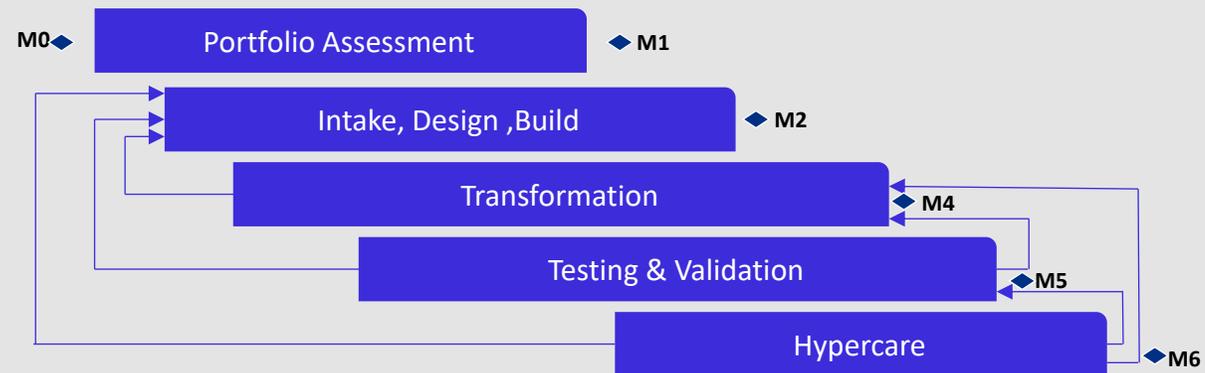
Workflow Modernization Beyond Translation
Reimagine workflows leveraging Airflow's dynamic DAG capabilities to enable event-driven, scalable, &self-healing orchestrations.

Augmented Rollout
40–60% faster rollout of new or modified workflows through DAG-as-code and CI/CD automation

Business Outcomes
' 30–50% reduction in total scheduling and orchestration TCO by eliminating proprietary licensing and optimizing infrastructure

The transformational Journey with Hexaware

Powered by amaze®

M0 — Portfolio Assessment — M1
Intake, Design ,Build — M2
Transformation — M4
Testing & Validation — M5
Hypercare — M6

An Introduction to amaze® Assessment & Transformation Platform

Our patented, home-grown purpose-built accelerator solution that helps in tech debt elimination

For Infrastructure | For Application | For Middleware | For Data

TECHNOLOGY WE HELP MODERNIZE

← **ASSESSMENT** →

← **TRANSFORMATION** →

Technology logos
AutoSys, ASP.NET, Java EE, IBM DB2, MySQL, python, JBoss by Red Hat, IBM WebSphere, Microsoft SQL Server, JSP, TIBCO, ORACLE, Windows Forms, ORACLE Weblogic, SYBASE, .NET
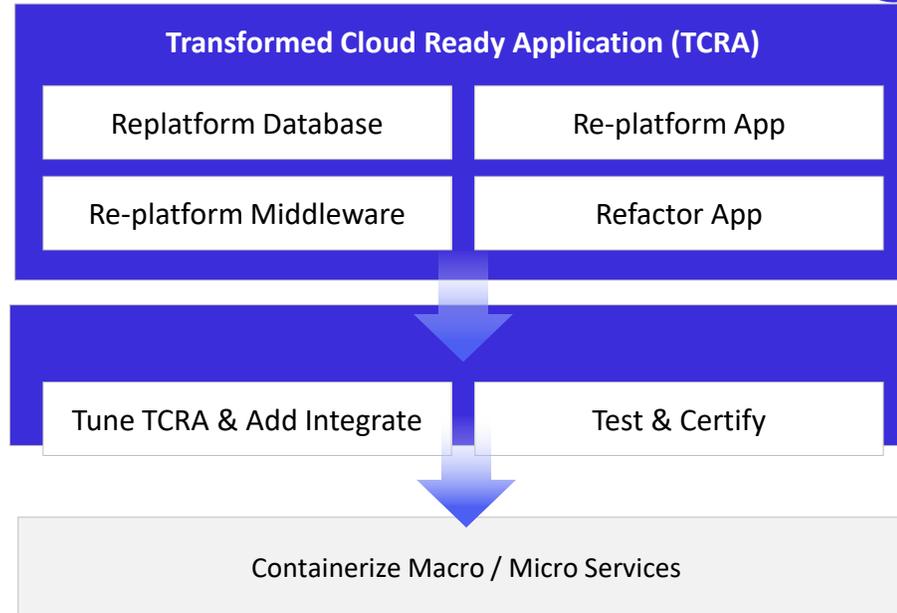
Assessment
- Transformation Readiness
- Transformation Type
- Transformation Complexity
- Transformation Effort

Transformation

App / DB Transformation

AUTOMATION **70%**

Transformed Cloud Ready Application (TCRA)

Replatform Database	Re-platform App
Re-platform Middleware	Refactor App

Tune TCRA & Add Integrate	Test & Certify

Containerize Macro / Micro Services

Metrics
90% Reduction in **SME time**

<1 Week To **Target State Architecture**

60% **Cost & Effort** Savings

10x Increase in **Success Rate**

Google Cloud | Microsoft Azure | aws | Private cloud

Business Benefits

- **Zero Impact to Business Functionality**
- **Decomposition** of Monolith resulting in improved **business agility and faster enhancement releases**
- Technology stack upgrade for Java & .NET resulting in **license remediation & technical debt remediation and reduced TCO**
- Database re-platforming from SQLServer, DB2, Sybase and Oracle to opensource PostgreSQL as part of **Remediation of license cost and divert the cost to business enhancements**
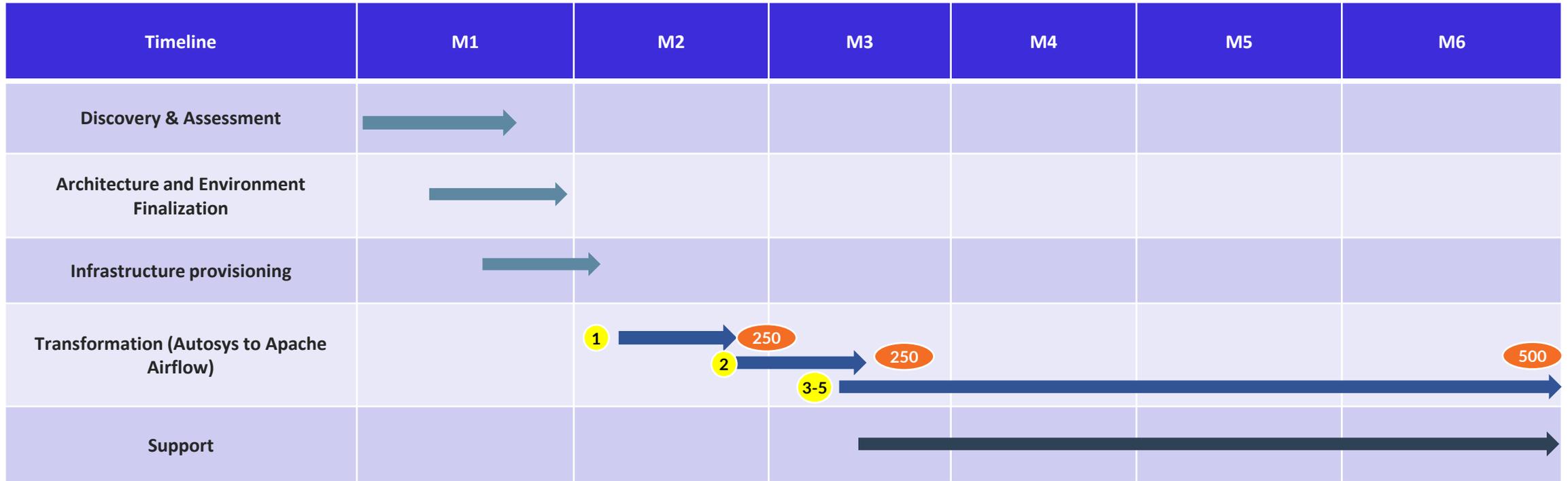
Autosys Transformation Journey

HEXAWARE

	Automated Assessment & Design	Batch Process Transformation (AMAZE-led)	Code Finalization & Validation	Airflow Onboarding & Go-Live
Activities	• Ingest JILs, scripts and configs • Parse jobs, calendars, dependencies & Identify common job patterns • Assess current vs target environment • Analyze embedded scripts and logic • Build workflow and dependency maps • Finalize security and deployment model	• Auto-convert jobs to Airflow DAGs • Translate schedules and dependencies • Convert scripts to Airflow tasks • Generate env-specific configurations • Flag complex jobs for manual fix	• Refine and modularize DAG code • Parameterize for environments • Enable CI/CD deployment pipeline • Unit and functional DAG testing • Validate SLAs and dependencies • Performance and stability testing	• Deploy DAGs to production • Configure monitoring and alerts • Execute phased cutover • Handover to operations team
Deliverables	• Complete job inventory • Dependency and impact report • Complexity classification • Mapping and design document • Future-state Airflow architecture	• Auto-generated Airflow DAGs • Converted schedules and dependencies • Transformation/reconciliation report • Exception list for refactoring	• Production-ready DAG codebase • Version-controlled components • Test and validation reports • SLA verification sign-off	• Live Airflow workflows • Monitoring and support runbooks • KT and operations documents • Post go-live optimization backlog

Recommended Autosys Elimination Model by Phase

HEXAWARE

Timeline	M1	M2	M3	M4	M5	M6
Discovery & Assessment						
Architecture and Environment Finalization						
Infrastructure provisioning						
Transformation (Autosys to Apache Airflow)	1	2 → 250	250 / 3-5			500
Support						

Legend:
- **#** Wave Number
- **#** Number of jobs migrated
- **Preparation**
- **Migrate & Test**
- **Support**

- Hexaware has assumed 1000 autosys jobs to derive the above wave groupings and Project Plan.

Why choose Hexaware?

Upto 70% automation with **Amaze®**

Amaze® integration with leading DevOps & CI/CD

Proven case studies in both regulated and non-regulated industries

Our expertise ensures compliance, security, and innovation tailored to your industry needs.

Dedicated **Centers of Excellence** for various hyperscalers

Accelerated cloud adoption and maximum value from hyperscaler investments.

Reusable DAG templates and operator frameworks

Accelerated development cycles, and reduced operational complexity

Approx. **1000** Autosys experts
11000+ certified cloud consultants

Hybrid Skillsets Across Job scheduling Automation & Microservices

500+ transformations executed
1400+microservices generated

scalability, flexibility, and faster time-to-market

Price Comparison

HEXAWARE

Cost Category	Using Autosys (Annual)	Using Airflow (Annual)
Operational Cost	High licensing fees (~$120,000+)	Minimal licensing (open-source)
Maintenance Cost	15-20% of licensing fees (~18000−24,000)	Lower maintenance (~12000−18,000)
Support Cost	Vendor support fees ~12000 - 18000	Internal or third-party support (~8000−12,000)
Transformation Cost	N/A (no transformation)	One-time cost: approx. 350,000
Infrastructure Cost	Moderate to High (proprietary hardware/software)	Flexible cloud/on-prem infrastructure, potentially lower cost
Total Estimated Annual Cost(approx.)	$150,000+	20000−30000 + transformation cost

Highlights

- For 1000 processes, the transformation cost = $350000

- Total cost for year1 – $370000-$380000(lesser than the total spend with TIBCO)

- From year 2 onwards: cost to company = $20000- $30000

- Cost savings for year 2 = $120000 - $130000(i.e , Full cost of transformation would be recovered within 3 year) based on cost savings

- ROI over 3 year is (~12%) as transformation cost is fully accounted.

- After the year 3 , the investment is recovered, and from year 4, the organization benefits from substantial cost savings, which will significantly increase ROI over time

Autosys to Apache Airflow Modernization for a Large Mortage Firm with 400000+ jobs and 126 applications

Business Problem:

The client relied on Autosys, a licensed, legacy workflow scheduler and orchestration tool. However, there were ample challenges posed by using autosys

- High licensing and maintenance costs (approx.950k $ annually , likely to increase upto 6M$- 20M $

- Not natively supported on AWS, lacking cloud-native orchestration capabilities.

- Complexity in embedded business logic and integration with 126 applications.

- Over 400,000 automated jobs critical for loan management and enterprise data infrastructure depended on Autosys.

- Existing contract with third-party reseller Kyndryl was at risk of non-renewal or steep price increases.

Solution:

Hexaware implemented migration from Autosys to cloud-native Apache Airflow deployed on AWS (Amazon Managed Workflows for Apache Airflow - MWAA).

- Utilized Hexaware's AI-powered AMAZE® tool for automated conversion of over 21,000 Autosys jobs per application to Airflow DAGs.

- Deployed infrastructure using Terraform automation, customized to client's environment

- Migration executed in phases:
  - 1-2 assets deployed per month
  - Converted 100-200 jobs per environment
  - Successfully transitioned 6 environments

- Comprehensive assessment and automation included:
  - Deep analysis of business logic, workflows, shell scripts.
  - Dependency mapping and job flow visualization.
  - Validation for no business functional impact.
  - UI-driven artifact generation, smoke testing, and go-live support, fully deployed in the cloud

50%
Reduction in TCO

50%
Faster migration

40%
Less implementation costs due to automation

You are Modernizing with Hexaware

Supercharge with Amaze

Activate Amaze automation within your environment to fast-track your transformation

Next Steps

Client

1. Shout out for discovery workshop
2. Access to autosys environment

Hexaware

1. Assessment of autosys workflows
2. Assessment Report
3. Migration POC for 100 workflows
4. Roadmap for complete migration

Contact Hexaware for a personalized consultation and roadmap to your cloud-native future.

Email :
Website
Ph:

Technical Workflow

Enterprises often use commercial schedulers & workflow management tools that are legacy in nature and do not align with modern IT environments & landscape. With AutoSys, organizations have created complex integrations and embedded business logic which presents numerous challenges in both understanding, planning and execution of Migration.

Business logic intertwined at **multiple layers** through out the workflow

Integration to multiple **Upstream & Downstream** applications often increasing the complexity in tracing

Shell Scripts which are part of the process flow create sub-flows within the business process

Custom error message formats & statuses result in **high learning curve** prior to migration

Business logic written across stored procedures & queries through direct integration with databases

Execution of the jobs are Dependent on the **status prior to kickstart**

Our Migration Approach

Source code Assimilation
Identification of all impacted source code files for thorough analysis & assessment

Deep Analysis
Assessment of business logic, workflow, integration & error handling

Simplified Transformation
Automated transformation to Fannie Mae approved Airflow solution for all components across jobs

Validation
Streamlined validation across current & post transformation to ensure no business functional impact

AWS Airflow MWAA System Overview Architecture
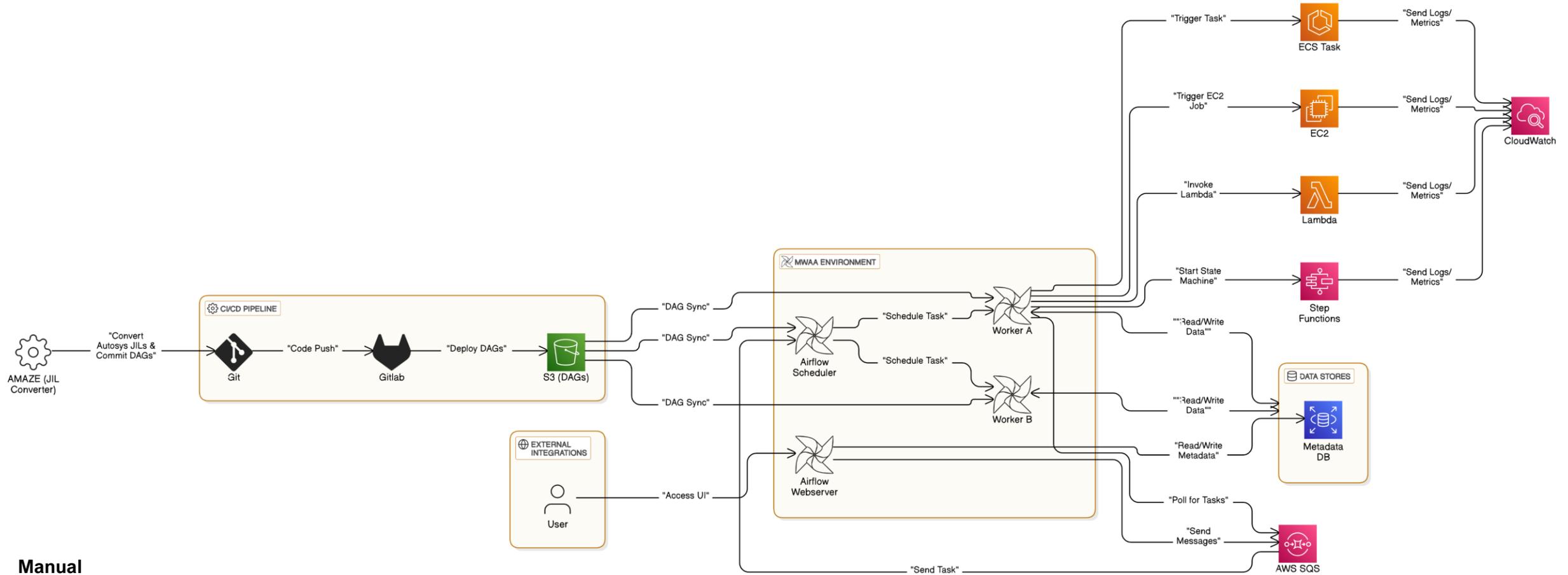
Manual

- Workflows are run locally as well as on AWS and debugged
- Workflow is executed using appropriate mechanisms depending on the job type
- Jobs can also be kickstarted from the Amazon MWAA Web UI
- The L1/L2 support teams use the UI to integrate with their current collaboration mechanisms in place with the business

AutoSys Assessment & Transformation Process Flow using AMAZE®

HEXAWARE

App Modernization Process
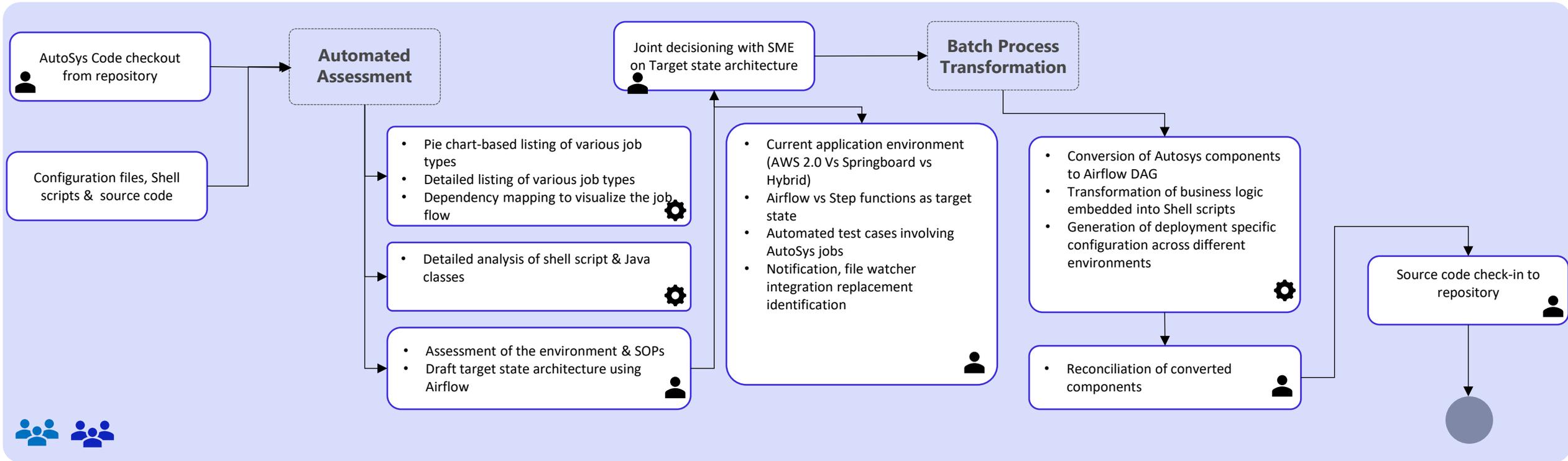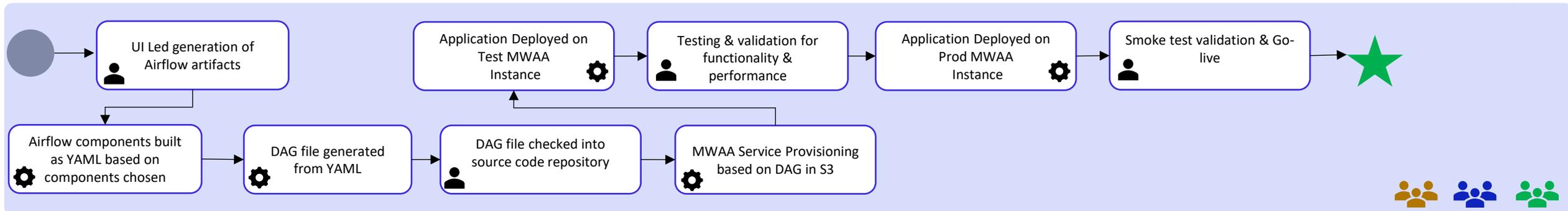
AutoSys Code checkout from repository

Configuration files, Shell scripts & source code

Automated Assessment

- Pie chart-based listing of various job types
- Detailed listing of various job types
- Dependency mapping to visualize the job flow

- Detailed analysis of shell script & Java classes

- Assessment of the environment & SOPs
- Draft target state architecture using Airflow

Joint decisioning with SME on Target state architecture

- Current application environment (AWS 2.0 Vs Springboard vs Hybrid)
- Airflow vs Step functions as target state
- Automated test cases involving AutoSys jobs
- Notification, file watcher integration replacement identification

Batch Process Transformation

- Conversion of Autosys components to Airflow DAG
- Transformation of business logic embedded into Shell scripts
- Generation of deployment specific configuration across different environments

- Reconciliation of converted components

Source code check-in to repository

Airflow Application Onboarding

UI Led generation of Airflow artifacts

Airflow components built as YAML based on components chosen

DAG file generated from YAML

DAG file checked into source code repository

MWAA Service Provisioning based on DAG in S3

Application Deployed on Test MWAA Instance

Testing & validation for functionality & performance

Application Deployed on Prod MWAA Instance

Smoke test validation & Go-live

Application Team   Hexaware AMAZE® Re-platform Team   Testing & Validation Team

15

AutoSys Job Type	Primary Complexity Factors	Migration Target	Base Complexity
Command (c)	- Data Volume & Time	Step Function (EC2 task)	Low (5-15)
	- Error Handling	MWAA (BashOperator)	
	- Cross-Platform Integration		
Box (b)	- Dependencies & Nesting	Step Function (Parallel/Sequential)	Medium-High (15-40)
	- Scheduling Complexity	MWAA (TaskGroup)	
	- Error Handling		
File Watcher (FW)	- Data Volume & Time	Step Function (Lambda + S3 Event)	Medium (10-25)
	- Error Handling	MWAA (FileSensor)	
	- Conditional Logic		
File Trigger (FT)	- Cross-Platform Integration	Step Function (EventBridge)	Medium (10-25)
	- Conditional Logic	MWAA (FileSensor + Trigger)	
	- Error Handling		
SQL Jobs	- Data Volume & Time	Step Function (Glue Job)	Medium (15-30)
	- Cross-Platform Integration	MWAA (SQLOperator/GlueOperator)	
	- SLA & Criticality		
Script Jobs (Shell/Python)	- Conditional Logic	Step Function (Lambda/ECS)	Low-Medium (10-25)
	- Error Handling	MWAA (BashOperator/PythonOperator)	
	- Cross-Platform Integration		

T-Shirt Sizing: Parameters for AutoSys Decommission

Category	Score Range	Characteristics	AutoSys Examples
Small (S)	0-35	- Simple command jobs - 1-2 dependencies - Basic scheduling - Minimal error handling - Single machine - No SLA	- Standalone command jobs - Simple file watchers - Basic shell scripts - Single box (1 level)
Medium (M)	36-65	- Box jobs (2-3 levels) - 3-8 dependencies - Calendar + time windows - Moderate error handling - 2-3 machines - Standard SLA	- Multi-step workflows - File trigger chains - SQL + data processing - Moderate nesting - Cross-app dependencies
Complex (C)	66-100	- Deep nesting (4+ levels) - 9+ dependencies - Complex calendars - Advanced error handling - 4+ machines/platforms - Strict SLA/Critical	- Enterprise workflows - Multi-app orchestration - Legacy integration - Regulatory compliance - High data volume