**ServiceNow Platform Analytics Migration Strategy**

**Executive Summary**

This document outlines a comprehensive strategy for migrating from Performance Analytics to Platform Analytics in ServiceNow. ServiceNow announced in February 2024 its plan to phase out the traditional ServiceNow Core UI reports and dashboards by its Zurich release in 2025, making this migration mandatory for all organizations.

T**imeline Critical**: With the Zurich release everything will be migrated (no way of opting out, Platform Analytics will be the only way to go)

**Project Overview**

**Key Drivers**

 **Mandatory Migration**: Platform Analytics becomes the default solution for organizations implementing ServiceNow in the Xanadu release

 **Enhanced Capabilities**: Modern interface with advanced data visualization and analysis tools

 **Performance Improvements**: Platform Analytics saves users three-quarters of the time required to access analytics

**Migration Scope**

 Legacy dashboards from Core UI  Performance Analytics widgets

 Reports and filters

 Custom visualizations and configurations

**Migration Strategy**

**Phase 1: Pre-Migration Assessment & Planning (Weeks 1-4)**

* 1. **Inventory and Assessment**

**Deliverables**: Complete inventory of current analytics assets **Activities**:

**Dashboard Audit**: Navigate to to catalog all existing dashboards

pa\_dashboards.list

**Report Analysis**: Document all reports, their usage frequency, and business criticality

 U**ser Access Review**: Map user roles, permissions, and access patterns

 **Performance Baseline**: Document current dashboard load times and performance metrics

 **Dependency Mapping**: Identify dashboards with embedded charts in forms, mobile usage, and Service Portal integrations

T**imeline**: 2 weeks

* 1. **Environment Preparation Activities**:

 Set up dedicated test environment (Sub-Production)  Configure Platform Analytics Migration Center access

 Remove inactive dashboards and reports to ensure a smooth migration process

 Backup current configurations and custom code T**imeline**: 1 week

* 1. **Risk Assessment**

**High-Risk Items to Identify**:

 Dashboards with complex custom logic

 Heavy usage dashboards with performance concerns  Integration points with external systems

 Custom widgets and advanced configurations

T**imeline**: 1 week

**Phase 2: Pilot Migration (Weeks 5-8)**

* 1. **Pilot Selection Strategy**

If you want to test with some teams the migration and the move to Platform Analytics then you can do a phased migration of just some dashboards

**Pilot Criteria**:

Select 5-10 representative dashboards

Include mix of simple and complex visualizations Choose dashboards from different business units Include both high and low usage dashboards

T**imeline**: 1 week

* 1. **Pilot Execution Process**:
     1. Navigate to Performance Analytics > Migration Center
     2. Use "Start Moving" for selected dashboards
     3. The system automatically identifies content compatible with the migration and performs the procedure
     4. Review items in "Compatibility Mode" T**imeline**: 1 week
  2. **Pilot** V**alidation &** T**esting** V**alidation Framework**:

 **Functional** T**esting**: Verify all visualizations render correctly

 **Data Accuracy**: Compare data between old and new dashboards  **Performance** T**esting**: Measure load times and responsiveness

 U**ser Acceptance**: Gather feedback from pilot users

 **Access Control**: Verify role-based permissions work correctly T**imeline**: 2 weeks

**Phase 3: Issue Resolution & Refinement (Weeks 9-12)**

* 1. **Known Migration** T**ool Issues**

Based on community feedback, the following issues have been identified: **Critical Issues**:

Existing breakdown source field is not being followed. The way these breakdown sources are being migrated the new filters are not showing the same elements as the pre-migration breakdowns

Column visualizations that have been migrated from an existing PA Widget. The indicator list order is not being kept and the order needs to be reset

Single score format values are not being migrating with the same specs as those that are contained within the indicator

**Functional Gaps**:

 Chart type reports on dashboards that display data tables after the chart lose this functionality in migrated versions

 Knowledge articles displayed as iframes become difficult to read in migrated dashboards, formatted as small squares with scroll bars

 Cascading Filters were previously neatly packaged and now these are separated

* 1. **Issue Resolution Strategy Approach**:
     1. **ServiceNow Support Cases**: Please raise Support cases if you see issues and we will be able to verify directly with the internal team
     2. **Custom Development**: Plan for Java Spring Boot APIs to handle data gaps if needed
     3. W**orkaround Documentation**: Document alternative approaches for unsupported features
     4. U**ser** T**raining**: Prepare training for new UI patterns and workflows T**imeline**: 4 weeks

**Phase 4: Bulk Migration (Weeks 13-16)**

* + - 1. **Pre-Migration Cleanup Activities**:

 Final removal of unused/inactive dashboards  Consolidate duplicate reports

 Update documentation and user guides

 Communication to end users about upcoming changes T**imeline**: 1 week

* + - 1. **Bulk Migration Execution Process**:
         1. Schedule maintenance window
         2. Execute bulk migration through Migration Center
         3. The migration is the one that migrates all content
         4. Monitor migration progress and log any failures
         5. After bulk migration, all the URL that were going to Core UI artifacts will automatically redirect the users to the migrated version

T**imeline**: 1 week

* + - 1. **Post-Migration** V**alidation Comprehensive** T**esting**:

 **Data Integrity**: Automated scripts to compare data accuracy

 **Performance Benchmarking**: Compare against baseline metrics  U**ser Access** T**esting**: Validate all role-based permissions

 **Integration** T**esting**: Verify embedded charts and mobile functionality  **Load** T**esting**: Test with peak user volumes

T**imeline**: 2 weeks

**Phase 5: Optimization & User Adoption (Weeks 17-20)**

* 1. **Performance Optimization Activities**:

 Review dashboard caching settings (available only in Platform Analytics)  Optimize slow-performing visualizations

 Clean up redundant indicator sources

 Implement best practices for data source selection

* 1. U**ser** T**raining & Support** T**raining Program**:

 Create new user guides reflecting Platform Analytics interface  Conduct training sessions for power users

 Set up help desk support for migration-related questions  Document differences between old and new systems

* 1. **Ongoing Monitoring Monitoring Strategy**:

Track dashboard usage statistics

 Monitor performance metrics

 Collect user feedback continuously  Plan for iterative improvements

T**imeline**: 4 weeks

**Post-Migration** V**alidation Framework**

**Data Validation Steps**

1. **Automated Data Comparison**

java

*// Example validation approach using Spring Boot* @Service

public class MigrationValidationService {

public ValidationResult validateDashboardData(String dashboardId) {

*// Compare data sources between old and new dashboards*

*// Validate aggregation calculations*

*// Check date range filters*

*// Verify breakdown dimensions* return validationResult;

}

}

1. **Key** V**alidation Points**

 **Data Accuracy**: Row counts, sum totals, averages match between old and new  **Date Filtering**: Ensure date range selectors produce identical results

 **Breakdown Sources**: Verify dimension filtering works correctly

 **Aggregations**: Confirm sum, count, average calculations are accurate  **Drill-down Functionality**: Test navigation between dashboard levels

1. **Performance** V**alidation**

**Load** T**ime Comparison**: Document before/after load times **Concurrent** U**ser** T**esting**: Validate performance under load **Mobile Performance**: Test dashboard rendering on mobile devices **Memory** U**sage**: Monitor browser memory consumption

1. **Functional** V**alidation Checklist Dashboard Level**:

All widgets display correctly Dashboard filters function properly Date range selectors work as expected Interactive elements respond correctly Print/export functionality works

W**idget Level**:

Data visualization types render correctly Color schemes and formatting preserved Tooltips and hover effects work

Click-through actions function properly Data labels and legends display correctly

U**ser Access**:

Role-based permissions enforced Group-level restrictions maintained Personal dashboard access preserved Sharing settings migrated correctly

**Risk Mitigation & Contingency Plans**

**High-Risk Scenarios**

1. **Migration** T**ool Failures**

**Risk**: Critical functionality not migrated correctly **Mitigation**:

 Maintain parallel systems during transition period

 Develop custom React components for missing functionality  Contact NOW Support for rollback options if needed

1. **Performance Degradation**

**Risk**: Migrated dashboards perform worse than original **Mitigation**:

Pre-migration performance baseline establishment Dashboard caching implementation

 Query optimization and indicator source consolidation

1. U**ser Adoption Challenges**

**Risk**: Users struggle with new interface **Mitigation**:

 Comprehensive training program

 Side-by-side interface comparisons

 Gradual transition period with both systems available

**Rollback Strategy**

It is my understanding that there is a way to rollback the migration but you need to contact NOW Support in order to do so

**Rollback Conditions**:

 Critical functionality completely broken  Severe performance degradation

 Data integrity issues that cannot be resolved

**Rollback Process**:

1. Immediate escalation to ServiceNow Support
2. Document specific issues preventing production use
3. Coordinate rollback timing with business stakeholders
4. Plan for re-migration approach addressing identified issues

T**imeline Summary**

|  |  |  |
| --- | --- | --- |
| **Phase** | **Duration** | **Key Deliverables** |
| Pre-Migration Assessment | 4 weeks | Inventory, risk assessment, environment setup |
| Pilot Migration | 4 weeks | Pilot execution, validation framework, user feedback |
| Issue Resolution | 4 weeks | Bug fixes, workarounds, support cases |
| Bulk Migration | 4 weeks | Full migration, comprehensive validation |
| Optimization & Adoption | 4 weeks | Performance tuning, training, monitoring |
| **Total Project Duration** | **20 weeks** | **Fully migrated Platform Analytics implementation** |

**Success Criteria**

**Technical Success Metrics**

 100% of critical dashboards migrated successfully

 No more than 10% performance degradation on any dashboard  All data validation tests pass with 99.9% accuracy

 Zero critical functionality gaps

**Business Success Metrics**

 User adoption rate >90% within 4 weeks post-migration

 Reduction in support tickets related to analytics within 8 weeks  Improved user satisfaction scores for analytics experience

 Maintained or improved dashboard usage statistics

**Recommendations**

**Immediate Actions**

1. **Start Planning Now**: Begin inventory and assessment immediately
2. **Environment Setup**: Establish dedicated test environment for migration testing
3. **Stakeholder Engagement**: Involve business users early in pilot testing
4. **Support Relationship**: Establish direct communication channel with ServiceNow Support

**Best Practices**

1. **Phased Approach**: Use phased migration for testing before bulk migration
2. **Comprehensive** T**esting**: Validate both functionality and performance thoroughly
3. U**ser Communication**: Keep users informed throughout the migration process
4. **Documentation**: Maintain detailed documentation of all changes and workarounds

**Long-term Considerations**

1. **Continuous Monitoring**: Implement ongoing performance monitoring
2. **Regular** U**pdates**: Stay current with Platform Analytics feature releases
3. U**ser Feedback**: Establish continuous feedback loop for improvements
4. **Optimization**: Plan for regular optimization reviews and updates

**Document** V**ersion**: 1.0

**Last** U**pdated**: January 2025

**Next Review**: Post-Migration (Week 21)