Phase 2C Removal Matrix

Overview

This document provides a comprehensive mapping of legacy components to their new Phase 2C equivalents, along with removal procedures and timelines.

Legacy → New Component Mapping

Configuration Files

Legacy Path	New Path	Status	Removal Date
`old_configs/ans-ible.cfg`	`ansible.cfg`	✅ Migrated	2025-10-15
`legacy_inventory/`	`environments/*/in-ventories/`	✅ Migrated	2025-10-15
`old_group_vars/`	`group_vars/`	✅ Migrated	2025-10-15
`deprec-ated_host_vars/`	`host_vars/`	✅ Migrated	2025-10-15
`old_vault_files/`	`group_vars/*/vault.yml`	✅ Migrated	2025-10-20

Playbooks

Legacy Playbook	New Playbook	Status	Removal Date
`deploy_old.yml`	`site.yml`	✅ Migrated	2025-10-10
`setup_monitoring_v1.yml`	`monitoring.yml`	✅ Migrated	2025-10-12
`secur-ity_hardening_old.yml`	`security.yml`	✅ Migrated	2025-10-12
`backup_legacy.yml`	`backup.yml`	✅ Migrated	2025-10-15
`maintenance_old.yml`	`maintenance.yml`	✅ Migrated	2025-10-15

Roles

Legacy Role	New Role	Status	Removal Date
`roles/old_webserver`	`roles/webserver`	✅ Migrated	2025-10-10
`roles/leg-acy_database`	`roles/database`	✅ Migrated	2025-10-10
`roles/monitoring_v1`	`roles/monitoring`	✅ Migrated	2025-10-12
`roles/security_old`	`roles/security`	✅ Migrated	2025-10-12
`roles/backup_legacy`	`roles/backup`	✅ Migrated	2025-10-15
`roles/logging_v1`	`roles/logging`	✅ Migrated	2025-10-15
`roles/load_balancer_old`	`roles/load_balancer`	✅ Migrated	2025-10-18

Scripts

Legacy Script	New Script/Make Target	Status	Removal Date
`scripts/old_deploy.sh`	`make deploy-{env}`	✅ Migrated	2025-10-08
`scripts/leg-acy_backup.sh`	`make backup`	✅ Migrated	2025-10-10
`scripts/old_lint.sh`	`make lint`	✅ Migrated	2025-10-08
`scripts/secur-ity_scan_old.sh`	`make security-test`	✅ Migrated	2025-10-12
`scripts/monitor-ing_check_v1.sh`	`make monitor-valid-ate`	✅ Migrated	2025-10-15

Documentation

Legacy Document	New Document	Status	Removal Date
`OLD_README.md`	`README.md`	✅ Migrated	2025-10-05
`docs/old_architecture.md`	`docs/ARCHITEC-TURE.md`	✅ Migrated	2025-10-08
`legacy_deployment_guide.md`	`docs/deployment_guide.md`	✅ Migrated	2025-10-10
`old_troubleshooting.md`	`docs/troubleshooting_guide.md`	✅ Migrated	2025-10-12
`monitoring_guide_v1.md`	`docs/monitoring_guide.md`	✅ Migrated	2025-10-15

CI/CD Pipelines

Legacy Pipeline	New Pipeline	Status	Removal Date
`.github/workflows/old_ci.yml`	`.github/workflows/ci.yml`	✅ Migrated	2025-10-05
`.gitlab-ci-old.yml`	N/A (Removed)	✅ Removed	2025-10-05
`jenkins/old_pipeline.groovy`	N/A (Removed)	✅ Removed	2025-10-05

Environment Configurations

Legacy Environment	New Environment	Status	Removal Date
`envs/dev_old/`	`environments/dev/`	✅ Migrated	2025-10-08
`envs/test_legacy/`	`environments/test/`	✅ Migrated	2025-10-08
`envs/prod_v1/`	`environments/prod/`	✅ Migrated	2025-10-10
`staging_old/`	`environments/staging/`	✅ Migrated	2025-10-10

Removal Procedures

Automated Cleanup

The following components will be automatically removed by the cleanup automation:

Phase 1: Safe Removals (Completed)

- ✅ Unused configuration files
- ✅ Deprecated documentation
- ✅ Old CI/CD pipelines
- ✅ Legacy scripts with direct replacements

Phase 2: Validated Removals (In Progress)

- 🔄 Legacy roles (after validation)
- 🔄 Old playbooks (after testing)
- 🔄 Deprecated inventory files
- 🔄 Legacy environment configurations

Phase 3: Final Cleanup (Scheduled)

- ⏳ Archive directories
- ⏳ Backup legacy configurations
- ⏳ Remove temporary migration files
- ⏳ Clean up git history (optional)

Manual Validation Required

The following components require manual validation before removal:

Component	Validation Required	Assigned To	Due Date
`roles/custom_legacy`	Functionality verification	DevOps Team	2025-10-20
`scripts/custom_deploy.sh`	Integration testing	Platform Team	2025-10-18
`legacy_monitoring/`	Metrics validation	SRE Team	2025-10-22
`old_security_configs/`	Security review	Security Team	2025-10-25

Rollback Procedures

In case issues are discovered after removal:

1. **Immediate Rollback**
   ```bash
   git revert <removal_commit>
   git push origin main
   ```

2. **Restore from Backup**

```bash
   # Restore from backup archive
   tar -xzf legacy_backup_$(date +%Y%m%d).tar.gz
   # Validate restoration
   make gate-integration
```

3. **Emergency Recovery**
   - Contact on-call engineer
   - Follow incident response procedures
   - Restore from last known good state

Migration Validation

Pre-Removal Checklist

For each component scheduled for removal:

- [ ] New component functionality verified
- [ ] All dependencies updated
- [ ] Documentation updated
- [ ] Team training completed
- [ ] Rollback procedure tested
- [ ] Stakeholder approval obtained

Post-Removal Validation

After removal:

- [ ] All quality gates pass
- [ ] Golden path tests successful
- [ ] Performance benchmarks met
- [ ] Security scans clean
- [ ] Monitoring alerts normal
- [ ] No broken dependencies

Communication Plan

Stakeholder Notifications

Stakeholder Group	Notification Method	Timeline
Development Teams	Slack + Email	1 week before removal
Operations Teams	Email + Meeting	2 weeks before removal
Security Team	Direct communication	1 week before removal
Management	Status report	Weekly updates

Documentation Updates

- [ ] Update README.md with removal notices
- [ ] Update deployment guides
- [ ] Update troubleshooting documentation
- [ ] Update training materials
- [ ] Archive legacy documentation

Risk Assessment

High Risk Removals

Component	Risk Level	Mitigation
`roles/database_legacy`	🔴 High	Extended testing period
`scripts/critical_backup.sh`	🔴 High	Manual validation required
`legacy_monitoring/alerts`	🟡 Medium	Gradual migration

Low Risk Removals

Component	Risk Level	Status
`old_documentation/`	🟢 Low	✅ Completed
`deprecated_configs/`	🟢 Low	✅ Completed
`unused_scripts/`	🟢 Low	✅ Completed

Cleanup Automation Configuration

Automated Cleanup Settings

```yaml
cleanup_automation:
  enabled: true
  dry_run: false
  schedule: "0 2 * * 0"  # Weekly on Sunday at 2 AM
  retention_days: 30
  backup_before_removal: true
  notification_channels:
    - slack: "#infrastructure"
    - email: "devops@company.com"
```

Manual Override

To disable automated cleanup:

```
# Disable cleanup automation
export CLEANUP_AUTOMATION_ENABLED=false

# Enable manual mode only
export CLEANUP_MANUAL_MODE=true
```

Progress Tracking

Overall Progress

- **Total Components**: 45
- **Migrated**: 38 (84%)
- **In Progress**: 5 (11%)
- **Pending**: 2 (5%)

By Category

Category	Total	Migrated	In Progress	Pending
Configuration	8	8	0	0
Playbooks	6	6	0	0
Roles	12	10	2	0
Scripts	8	6	2	0
Documentation	7	5	1	1
CI/CD	4	3	0	1

Timeline

```
Phase 2C Removal Timeline
├── Week 1 (Oct 1-7)    ✅ Configuration files
├── Week 2 (Oct 8-14)   ✅ Basic playbooks & scripts
├── Week 3 (Oct 15-21)  🔄 Roles & advanced scripts
├── Week 4 (Oct 22-28)  ⏳ Final validation & cleanup
└── Week 5 (Oct 29+)    ⏳ Archive & documentation
```

Success Criteria

Completion Criteria

- [ ] All legacy components removed or archived
- [ ] No broken dependencies
- [ ] All quality gates passing
- [ ] Documentation updated
- [ ] Team training completed
- [ ] Rollback procedures tested

Performance Criteria

- [ ] No performance degradation
- [ ] All SLOs maintained
- [ ] Deployment times improved
- [ ] Reduced maintenance overhead
- [ ] Improved security posture

Last Updated: September 26, 2025
Next Review: October 15, 2025
Owner: DevOps Team
Approver: Infrastructure Lead