



# RBDMS

---

11/13/2018

IMPLEMENTING RBDMS 3.0 ACROSS STATES

- 
- RBDMS.org
  - GitHub
  - Sharepoint
  - Sharefile
  - National (static) forms library
  - State inventory
  
  - Next Steps

# Agenda

---

- Primary Objectives
- Key Considerations
- Implementation Approach at a Glance

# RBDMS

---

## PRIMARY OBJECTIVES

# Primary Objectives

---

- Provide States a cost effective approach for implementing RBDMS 3.0 and fully leveraging its core capabilities and services
- Allow States to implement customizations without losing the ability to apply RBDMS 3.0 core updates as desired
- Create and grow a library of Online Forms, Rules, and Workflows to be re-used or serve as templates across State implementations
- Reduce code branching and enable states/contractors to reuse code from other projects
- Decrease the cost of development, implementation and ongoing maintenance

# RBDMS

---

## KEY CONSIDERATIONS

# Key Considerations

---

- Implement State customizations without changing the RBDMS 3.0 core
- Use GWPC GitHub for source code management and project collaboration
- Discussion: GitHub policies
  - when to check in code to GitHub
  - What to check in
- Conduct periodic code reviews with GWPC
- Promote code re-use
- Leverage existing RBDMS 3.0 coding practices and patterns
- Communicate RBDMS 3.0 core bugs and enhancement recommendations to GWPC via GitHub or email

# RBDMS

---

IMPLEMENTATION APPROACH AT A GLANCE

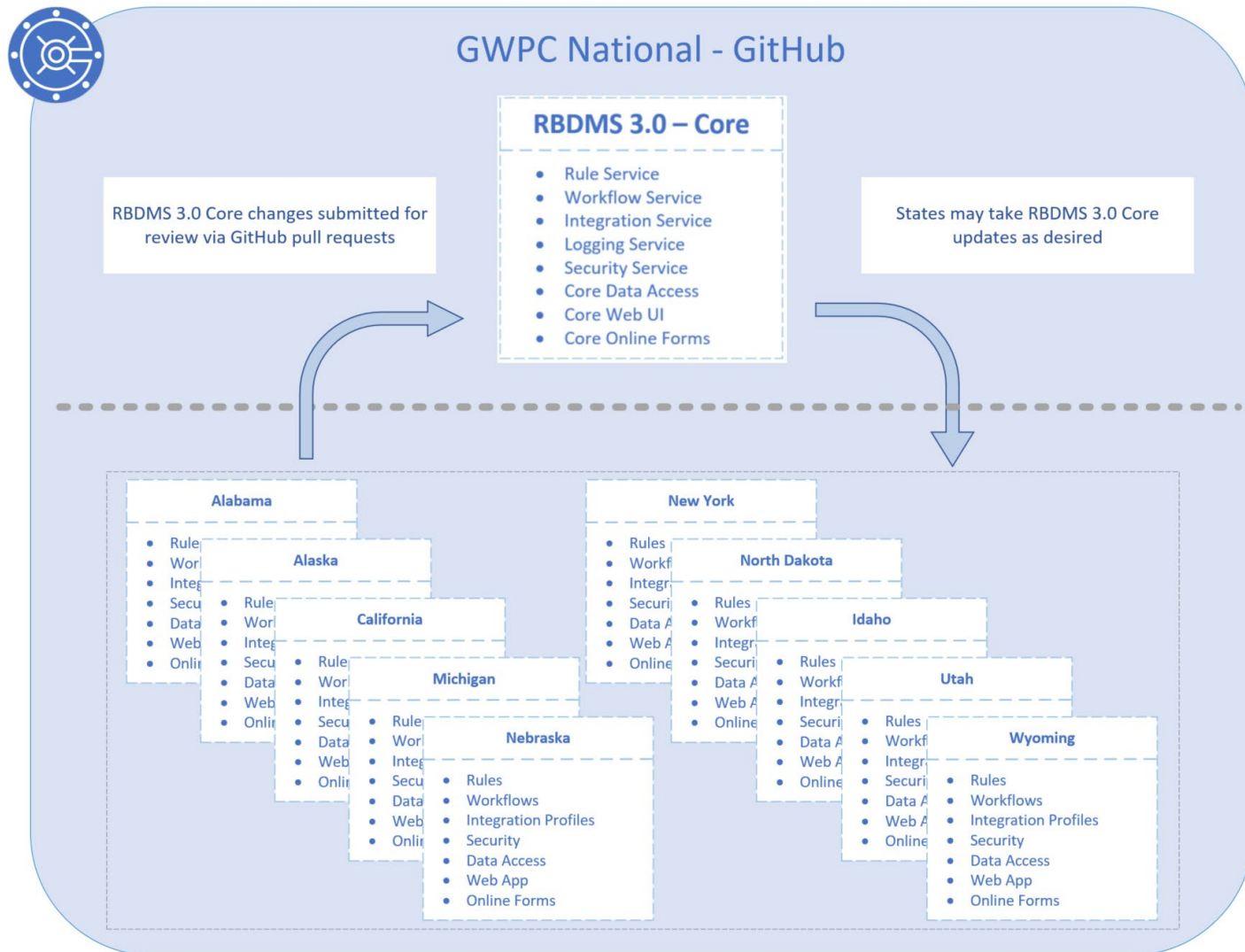


# Implementation Approach at a Glance

---

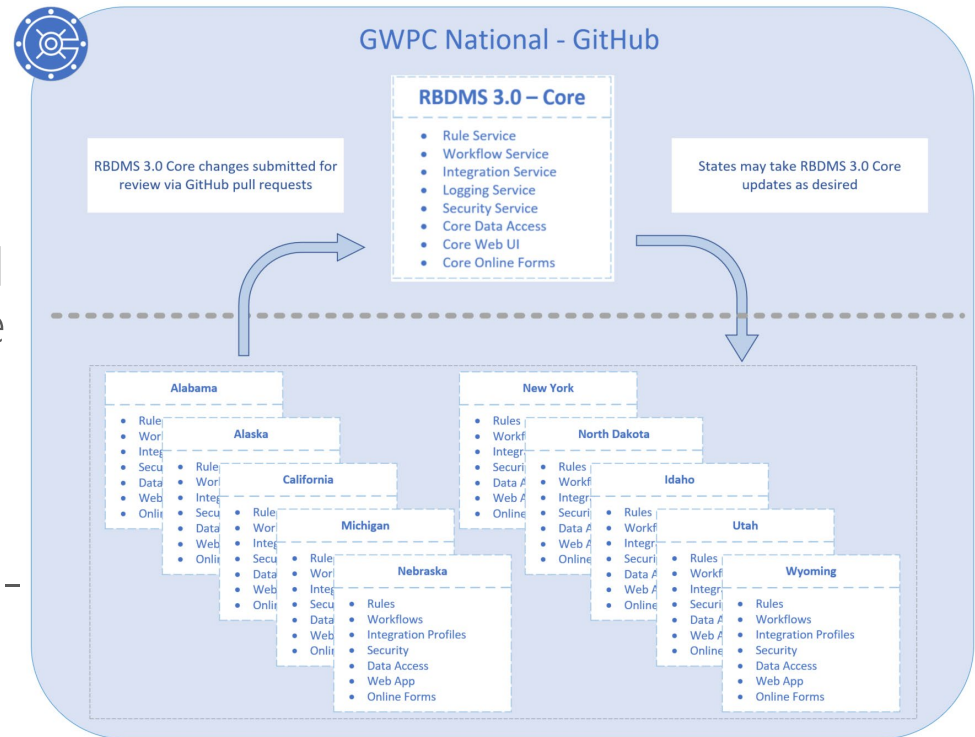
- Create GWPC GitHub repository for State project
- Create State RBDMS 3.0 Visual Studio solution to host application code where customizations are implemented
- Apply State branding
- Build and configure State backend access logic to support legacy data sources such as RBDMS Classic or .NET
- Install and configure RBDMS 3.0 core applications and services such as Rules and Workflow
- Design, Build, Test, and Deploy State implementation of RBDMS: 3.0, .net, Ohio
- Periodically apply RBDMS 3.0 core updates as desired

# Implementation Approach at a Glance... illustrated



# Implementation Approach at a Glance... described

- RBDMS 3.0 Core maintained in GitHub
- RBDMS 3.0 State implementations, including customizations, maintained in GitHub but separate from the core
- States may periodically take RBDMS 3.0 core updates as desired
- States may collaborate to share or re-use State customizations such as Online Form sections, Rules, and Workflows
- Through GitHub pull requests (or simple email), States may submit RBDMS 3.0 core bug fixes, enhancement requests, or component re-use candidates (e.g. common form section), for consideration by GWPC



# GWPC GitHub Policy Guidelines

---

- Maintain application source code and relevant artifacts in GWPC's GitHub repositories
- Keep source code for applications deployed to Production up to date in the Master branch
- Keep source code for applications deployed to Test up to date in its applicable branch
- Maintain active development source code in branches named to indicate development, For example, ND-dev-R3.0.0-R1.0.
- During active development, check-in source code on a frequency no less than weekly
- Upon each production release, create a corresponding GitHub repository Release package with relevant deployment artifacts
- Use GitHub projects to communicate and track bugs and potential feature enhancements

# Questions

---

