**Option: Update RSAM to Handle Country-Specific Access Controls**

**✅ What Can Be Done in RSAM?**

1. **Create Custom Role(s):**  
   Define roles like Global\_Read\_India, Global\_Read\_Malaysia, etc., tied to specific countries/data domains.
2. **Add Country-Specific Approval Steps in Workflows:**  
   Introduce workflow logic that routes access requests for certain countries to local data stewards/owners.
3. **Dynamic Rule Enforcement:**  
   Based on request attributes (like business unit, location, or data region), dynamically assign approval paths.
4. **Create Country Metadata in Entitlements:**  
   Update entitlements to carry metadata like country\_scope, and filter or validate based on that.
5. **Policy Mapping Layer in RSAM:**  
   Build a small policy mapping module in RSAM where entitlements are grouped and assigned based on country rules.

**⚠️ Complexities in RSAM Enhancement**

| **Complexity** | **Details** |
| --- | --- |
| 🧱 **RSAM Customization Restrictions** | RSAM is often a managed system; deep structural changes can be expensive and break future upgrades or support. |
| 🔀 **Workflow Logic Explosion** | Adding country-based flows increases the number of branches in approval logic – making workflows harder to maintain. |
| 🔒 **Compliance & Audit Risks** | If roles are not clearly documented or mapped to controls, it can cause SOX/ISO audit exceptions. |
| 🐢 **Longer Approval Timelines** | More steps in workflow = slower approval = business friction. |
| 🧩 **Integration Complexity** | If RSAM roles need to sync to downstream systems (like AD or cloud IAM), additional mapping or code is required. |
| 📈 **Scalability Issues** | As new countries or data domains get added, the role count grows — leading to sprawl and confusion. |

**🔄 Alternative Middle Ground with Lower Complexity**

**🟡 Option: Minimal RSAM Enhancement + System-Level Controls**

| **Feature** | **How It Works** | **Effort** | **Benefit** |
| --- | --- | --- | --- |
| **Tag entitlements with countries (metadata only)** | Use RSAM fields to tag entitlements like “India-Data”, “MY-Access” | Low | Doesn't disturb existing workflows |
| **Use System-Level Country Check** | Build logic in downstream app or identity system to validate access country tags | Medium | Keeps RSAM untouched but validates access |
| **Log-based Backward Audit** | Whenever country access happens, log who accessed what, which country, via what entitlement | Medium | Helps satisfy compliance without policy explosion |

**🔄 If You Still Want to Update RSAM… Here's a Design Path:**

**✅ Step-by-Step Plan:**

1. **Add Country Field to Entitlement Records**
2. **Update Workflow Rules to Check for Country Flags**
3. **Map Country -> Approver Group (use config/lookup table)**
4. **Update Access Request Form:**
   * Include purpose of access
   * Auto-suggest approvers based on selected country
5. **Create Custom Role Matrix for Common Global Roles**
6. **Sync with Downstream IAM Tools (AD, Okta, etc.)**
7. **Define Audit Reports to Track Country-Level Access**

**👀 Real-World Reference Cases**

| **Company** | **What They Did** | **Tool** |
| --- | --- | --- |
| **HSBC** | Created country-based entitlements but approvals were done in ServiceNow with RSAM integration | RSAM + SNOW |
| **Pfizer** | Used metadata on access roles and had a microservice layer enforcing dynamic access rules based on country | Homegrown + SailPoint |
| **Standard Chartered** | Used conditional approvals in RSAM, but later moved to central policy enforcement in Azure AD for performance | RSAM + Azure AD |

**✅ Summary: Should You Update RSAM?**

| **Criteria** | **Recommended Action** |
| --- | --- |
| If RSAM is managed externally | Avoid deep changes; use system-level controls |
| If your team has RSAM dev access & time | Add metadata and approval paths, not role sprawl |
| If access is critical & audited | Create a hybrid model: country tagging in RSAM + validation in downstream systems |
| If timelines are strict | Keep RSAM untouched, use external logic and logs |

**Comparison Table: RSAM Update vs. External Country Check**

| **Aspect** | **Enhancing RSAM Workflows** | **External Validation (System-level)** |
| --- | --- | --- |
| **Complexity** | High – needs dev changes, workflow logic updates | Medium – handled in external system or service layer |
| **Time to Implement** | Medium to Long (2–6 weeks depending on vendor dependency) | Short (1–2 weeks for basic implementation) |
| **Future Country Scalability** | Medium – requires workflow/role additions | High – add country in config or DB |
| **Governance & Compliance** | Strong – approval trail in RSAM | Needs external audit logs |
| **RSAM Maintenance Impact** | High – risk of regression or conflicts in upgrades | None – RSAM remains unchanged |
| **Audit Visibility** | Directly visible in RSAM | Must integrate with reporting system |
| **Flexibility** | Limited – every country logic must be baked into RSAM | High – dynamic logic with metadata/configs |
| **User Experience** | Consistent – all in RSAM | Split experience (request in RSAM, check outside) |
| **Example Use Case** | Malaysia data access routes to Malaysia approver | System checks if user’s role matches India\_Data\_Group |
| **Best For** | Highly regulated environments with centralized IAM | Agile, fast-scaling environments, low RSAM change control |

**🧠 Recommendation:**

* **For quick rollout and minimal disruption:**  
  → Go with **external country validation logic** (via service/middleware) + **audit logging**.
* **For long-term governance and audit clarity:**  
  → Use **RSAM metadata tagging** and **approval group mapping** via lookup/config table.
* **Avoid static approval workflows or new roles for every country** — they **don’t scale** and will become a maintenance burden.